



Escola d'Enginyeria de Telecomunicació i  
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# TREBALL DE FI DE CARRERA

**TÍTOL DEL TFC:** Desenvolupament d'eines software en Python per la generació automatitzada d'informes científics i tècnics d'instrumentació oceanogràfica

**TITULACIÓ:** Enginyeria Tècnica de Telecomunicació, especialitat Sistemes de Telecomunicació

**AUTOR:** Lorena Laguarda Rodríguez

**DIRECTOR:** Nuria Pujol Vilanova

**SUPERVISOR:** Lluís Alonso Zararte

**DATA:** 19 Febrer 2013



**Títol:** Desenvolupament d'eines software en Python per la generació automatitzada d'informes científics i tècnics d'instrumentació oceanogràfica

**Autor:** Lorena Laguarda Rodríguez

**Director:** Nuria Pujol Vilanova

**Supervisor:** Lluís Alonso Zararte

**Data:** 19 de febrer de 2013

## Resum

Actualment, l'observació de l'oceà apareix de manera prominent en els principals programes internacionals d'investigació i gestió ambiental. L'oceà té un gran potencial com a proveïdor de serveis energètics, i recursos naturals, i ocupa un paper central en la regulació del clima i dels cicles biogeoquímics globals. Es fa evident doncs, la necessitat de monitorar els processos que hi tenen lloc a través de l'adquisició de dades pel seu posterior processat.

L'ús de plataformes per a l'adquisició de dades ha anat evolucionant al llarg dels anys i oferint noves possibilitats . En l'actualitat, els vehicles autònoms submarins (AUV) són els que tenen més possibilitats en aquest camp, ja que gràcies a les seves característiques i les reduïdes dimensions, ofereixen una gran capacitat per adaptar-se al medi i accedir a entorns que d'una altra manera seria impossible explorar.

Dintre d'aquest marc localitzem l'objectiu principal del projecte, que consisteix en el disseny i desenvolupament d'una eina software per a la gestió i control de dades, i la generació automatitzada d'informes científics i tècnics a partir de les dades obtingudes en campanyes oceanogràfiques de vehicles autònoms submarins.

El projecte neix de la col·laboració amb la Unitat de Tecnologia Marina (UTM) del Centre Mediterrani d'Investigacions Marines i Ambientals (CMIMA-CSIC) i s'engloba dins de les activitats d'aquest grup i les noves línies de treball I+D relacionades amb els propis vehicles . El focus d'aquestes investigacions està ficat en la integració de nous sensors miniaturitzats, i la gestió i control de la qualitat de dades.

L'aplicació desenvolupada constituirà una eina *free software* que podrà ser utilitzada lliurement per la comunitat científica.

**Title:** Desenvolupament d'eines software en Python per la generació automatitzada d'informes científics i tècnics d'instrumentació oceanogràfica

**Author:** Lorena Laguarda Rodríguez

**Director:** Nuria Pujol Vilanova

**Date:** February, 19th 2013

## Overview

Nowadays, ocean observation appears prominently in major international research programs and environmental management. The ocean has a great potential as a supplier of energy services and natural resources, and occupies a central role in regulating climate and global biogeochemical cycles. The need to monitor the processes taking place through the acquisition of data for subsequent processing becomes thus clear.

Besides, the use of platforms for data acquisition has evolved over the years, offering many new possibilities. Currently, autonomous underwater vehicles (AUV) are the main ones in this area. Thanks to its characteristics, in term of features and small sizes, they offer a high capacity to adapt and access to environments, which otherwise it would be impossible to explore.

Considering this context, the project main objective consist in design and develop software tools to data management and control quality, and the automated generation of technical and scientific reports.

This project born in collaboration with the 'Unitat de Tecnologia Marina' of 'Centre Mediterrani d'Investigacions Marines i Ambientals' (CMIMA-CSIC) under activities of this research group and several new R+D lines, related with the vehicles themselves. The focus is put on small payload integration, data management (reporting, data archiving, metadata integration, etc..) and data Quality Control.

The developed software will be a free software tool available to scientific community.

# ÍNDEX

<b>INTRODUCCIÓ .....</b>	<b>1</b>
<b>CAPÍTOL 1. CONTEXT DEL PROJECTE .....</b>	<b>3</b>
1.1 Vehícles autònoms submarins en oceanografia .....	3
1.2 Objectius del projecte .....	5
1.3 Interès científic de l'aplicació.....	6
1.4 Software oceanogràfic .....	8
<b>CAPÍTOL 2. ENTORN DE DESENVOLUPAMENT .....</b>	<b>14</b>
2.1 Característiques dels vehicles .....	14
2.2 Batimetria i ADCP .....	15
2.2.2 AUV/DVL-ADCP .....	16
2.2.3 Batimetria a partir del DVL-ADCP .....	17
2.3 Eines software i llenguatge de programació .....	20
2.3.1 Software lliure .....	20
2.3.2 Llenguatge Python.....	21
2.3.3 Eines software de desenvolupament .....	21
<b>CAPÍTOL 3. DISSENY SOFTWARE .....</b>	<b>24</b>
3.1 Desenvolupament del software.....	24
3.2 Anàlisi dels requeriments .....	26
3.3 Arquitectura del software .....	27
3.4 Disseny del software .....	30
3.4.1 Mòduls per al tractament de les dades .....	30
3.4.2 Interfície gràfica (GUI) .....	36
3.5 Diagrama de classes UML .....	38
<b>CAPÍTOL 4. RESULTATS .....</b>	<b>41</b>
4.1 Aspectes generals de l'aplicació .....	42
4.2 Funcionalitats: .....	43
4.2.1 Monitoratge de variables .....	43
4.2.2 Generació d'estadístiques de campanya .....	44
4.2.3 Generació automatitzada de reports científics i tècnics .....	45
4.2.4 Batimetria del terreny .....	47
<b>CAPÍTOL 5. CONCLUSIONS .....</b>	<b>49</b>

5.1	Futures millores.....	50
	<b>BIBLIOGRAFIA .....</b>	<b>52</b>
	<b>ANNEX A: MANUAL D'USUARI .....</b>	<b>54</b>
	<b>ANNEX B: REPORT TÈCNIC.....</b>	<b>67</b>

# ÍNDIX DE FIGURES

Fig. 1.1 Imatge esquerra: ROV, imatge dreta: AUV .....	3
Fig. 1.2 Mapa de color de la profunditat sobre la trajectòria amb VectorMap ....	9
Fig. 1.3 Interfície EcoWatch: Representació de les variables dels sensors. ....	9
Fig. 1.4 Línies de contorn de profunditat i representació en 3D amb Surfer. ...	10
Fig. 1.5 Exemple de batimetria generada amb Fledermaus.....	11
Fig. 1.6 Interfície Ocean Data View.....	12
Fig. 2.1 Vehicles mini AUV configuració EcoMapper. ....	14
Fig. 2.2 Sistemes batimètrics; ecosondes monofeix i multifeix + GPS .....	16
Fig. 2.3 Profunditat ideal vs. profunditat real.....	17
Fig. 2.4 Posició d'intersecció i, j, k per als feixos 1-3 i 4-2 .....	18
Fig. 2.5 Sistema de coordenades del vehicle .....	19
Fig. 2.6 Qt Designer, exemple del procés de disseny d'un formulari.....	23
Fig. 3.1 Procés de desenvolupament model en V .....	25
Fig. 3.2 Estructura general del software.....	27
Fig. 3.3 Disseny arquitectura bloc estadístic i bloc gràfic .....	28
Fig. 3.4 Detall de l'arquitectura dels blocs report i batimetria .....	29
Fig. 3.5 Estructura de dades de tipus diccionari.....	30
Fig. 3.6 Entrada i sortida del mòdul stats.py.....	31
Fig. 3.7 Esquema d'entrades i sortides del mòdul report_pdf .....	33
Fig. 3.8 Il·lustració de la estructura de DocTemplate .....	34
Fig. 3.9 Fragment de codi de la funció scientific_stats .....	35
Fig. 3.10 Finestra de l'aplicació creada amb QtDesigner .....	37
Fig. 3.11 Diagrama de classes dels blocs Bathymetry, Plots i Statistics .....	39
Fig. 3.12 Diagrama UML bloc Report .....	40
Fig. 4.1 Finestra inicial de l'aplicació .....	42
Fig. 4.2 Simultaneïtat de funcions .....	42
Fig. 4.3 Exemple de diferents missatges d'alerta de auvSoft.....	43
Fig. 4.4 Finestra principal representació gràfica.....	44
Fig. 4.5 Interfície pel càlcul estadístic mostrant els resultats d'una missió. ....	45
Fig. 4.6 Assistant per a la creació d'un informe .....	46
Fig. 4.7 Configuració de la fitxa tècnica del report.....	47
Fig. 4.8 Interfície per generar la batimetria amb els resultats gràfics. ....	48
Fig. 4.9 Ampliació de la gràfica de resultats batimètrics.....	48
Fig. A.1 Pantalla del menú principal de l'aplicació.....	54
Fig. A.2 Visualització de les funcionalitats d'estadístiques i gràfiques .....	55
Fig. A.3 Pàgina 1 de l'assistant: Selecció del tipus d'informe.....	56
Fig. A.4 Pàgina 2 de l'assistant: Selecció dels fitxers.....	56
Fig. A.5 Pàgina 3 de l'assistant: Fitxa tècnica, introducció de metadades .....	57
Fig. A.6 Pàgina 4 de l'assistant: Dades estadístiques .....	57
Fig. A.7 Pàgina 5 de l'assistant: Selecció de les gràfiques d'usuari. ....	59
Fig. A.8 Pàgina 6 de l'assistant: Generació de l'informe. ....	59
Fig. A.9 Missatges finals del procés de report.....	60
Fig. A.10 Exemples d'informació del botó <i>Generate</i> (pas 6 de l'assistant).....	60
Fig. A.11 Diàleg de selecció dels fitxers.....	61
Fig. A.12 Pantalla principal del bloc d'estadístiques amb els resultats.....	61
Fig. A.13 Finestra per a la generació de la batimetria. ....	63
Fig. A.14 Propietats de la figura i zoom de la imatge .....	63
Fig. A.15 Formulari de selecció de variables.....	64

## LLISTA D'ACRÒNIMS

ADCP	Acoustic Doppler Current Profiler
AUV	Autonomous Underwater Vehicle
CSIC	Consejo Superior de Investigaciones Científicas
CMIMA	Centre Mediterrani d'Investigacions Marines i Ambientals
DVL	Doppler Velocity Log
GNU-GPL	General Public License
MIT	Massachusetts Institute of Technology
ODV	Ocean Data View
REMUS	Remote Environmental Monitoring UnitS
ROV	Remote Operated Vehicle
SPURV	Self-propelled underwater research vehicle.
UML	Unified Modeling Language
UTM	Unitat de tecnologia Marina
UTM	Universal Transverse Mercator
UUV	Unmanned undersea vehicles
WOCE	World Ocean Circulation Experiment
XML	Extensible Markup Language



# INTRODUCCIÓ

La importància del coneixement del mar ha anat incrementant-se en els últims anys, a mesura que nous descobriments han permès adquirir una ampla visió no tan sols de la seva significació present, sinó sobretot, futura.

Els satèl·lits es llencen a l'espai, a un ritme cada cop major, i a un cost que supera els cent milions d'euros. Sputnik va ser ficat en òrbita en 1957 i des de llavors, el món ha guanyat gran comprensió dels processos globals mitjançant l'ús de satèl·lits espacials. Què seria doncs, del nostre enteniment, si un nombre igual que el número de satèl·lits estiguessin explorant els nostres oceans *in situ*?

L'augment del coneixement del medi marí, des de les diferents vessants a les què satisfà l'oceanografia, no hauria sigut possible sense l'avenç tecnològic dels últims anys. Durant la dècada dels 90, s'han produït en tot el món, nombroses activitats d'investigació i desenvolupament en robòtica submarina, especialment en l'àrea dels vehicles autònoms submarins (AUV).

L'ús d'aquestes plataformes, obre noves possibilitats en les Ciències del Mar, ja que permet caracteritzar àrees relativament petites amb major resolució espacial i temporal que altres tècniques. Actualment, estem començant a veure com aquestes vehicles es converteixen en eines estàndard en molts països i cada cop més institucions científiques els adquireixen.

Entre aquests grups d'investigació es troba la Unitat de Tecnologia Marina del Centre Mediterrani d'Investigacions Marines i Ambientals UTM-CMIMA que recentment ha adquirit dos AUVs per incorporar-los als serveis que ofereix, amb l'objectiu de donar una nova perspectiva a la tradicional oceanografia.

És en aquest context on neix l'actual projecte, de la col·laboració, des de fa alguns anys, d'alumnes de la Universitat Politècnica de Catalunya amb el grup de recerca i desenvolupament de la Unitat de Tecnologia Marina CMIMA-CSIC i de l'interès d'aquesta per desenvolupar eines pròpies en el camp de la instrumentació i l'adquisició de dades en general. En concret aquest projecte té com objectiu el disseny i desenvolupament d'una eina software per a la gestió i control de dades, i la generació automatitzada d'informes científics i tècnics a partir de les dades obtingudes en campanyes oceanogràfiques de vehicles autònoms submarins.

L'estructura del document es divideix en cinc capítols dedicats a context del projecte, entorn de desenvolupament, disseny del software, resultats i conclusions.

El capítol de context del projecte fa una introducció als AUV i tracta tots aquells conceptes necessaris per tal de posar en situació al lector i definir els objectius a assolir. El capítol dedicat a l'entorn de desenvolupament, no només descriu els elements de hardware i software, sinó que també descriu els conceptes teòrics utilitzats en el desenvolupament, deixant exclusivament la part de

desenvolupament software englobada en el tercer capítol, on es detalla l'arquitectura del software dissenyat i els mòduls implementats. En el capítol 4, s'exposen els resultats obtinguts. Finalment s'exposen les conclusions obtingudes i les possibles millores.

# CAPÍTOL 1. CONTEXT DEL PROJECTE

## 1.1 Vehicles autònoms submarins en oceanografia

L'oceanografia, és la branca de les Ciències de la Terra, que estudia els processos biològics, físics, geològics i químics que es donen als mars i oceans. L'estudi integrat del medi marí és ja una necessitat inapel·lable que requereix de la convergència entre diferents disciplines d'investigació, tant bàsica com aplicada. En la banda de l'Oceanografia Operacional, s'estan imposant els vehicles autònoms submarins com plataformes submarines al servei de les Ciències del Mar, oferint una nova perspectiva a la tradicional oceanografia.

Els vehicles autònoms submarins (AUV) són robots submarins amb capacitat de moviment i desenvolupament de missions sota l'aigua, sense la necessitat de ser comandats per un operador. Disposen d'autonomia energètica i intel·ligència suficient com per dur a terme funcions preprogramades i operar allunyats de la seva plataforma de control, per tant precisen d'una important component *software* i *hardware* destinada a la navegació autònoma. Poden integrar una sèrie de sensors i mòduls personalitzats de càrrega útil que el fan flexible a les necessitats de l'usuari.

Els AUVs formen part d'un gran grup de sistemes submarins coneguts com Vehicles Submarins no tripulats (UUV), una classificació que inclou els populars Vehicles Operats a Distància (ROV). Aquets es diferencien dels AUVs perquè estan operats de forma remota a través d'un cable (cordó umbilical) que uneix el vehicle amb la seva plataforma de control. Al disposar de transmissió de potència des de la plataforma de control, el temps que pot romandre un ROV en operació és il·limitat, ja que no depèn d'una bateria. No obstant, el cable origina limitacions relacionades amb el radi d'abast en què pot operar el vehicle.



**Fig. 1.1** Imatge esquerra: ROV, imatge dreta: AUV

El desenvolupament dels AUVs va començar a inicis dels 60, amb vehicles com SPURV (Self-Propelled Underwater Research Vehicle), creat pel laboratori de física aplicada de la Universitat de Washington. Aviat el van seguir altres com SKAT (Russia), OSR-V (Japó), o els desenvolupats pel MIT al 1970.

La majoria d'aquests vehicles eren massa grans, ineficients i cars. Mentre els ROVs estaven començant a guanyar maduresa en la dècada dels 80, la tecnologia dels AUVs estava essencialment en la seva infantesa.

Inicialment doncs, degut a l'alt cost operatiu, l'aplicació d'aquest vehicles es reduïa al món militar per a missions de reconeixement, o cerca de mines. Però aviat els ROVS van trobar importants clients en la indústria del petroli i del gas, fent-se indispensables per a la inspecció, manteniment i reparació de cables, canonades, i altres estructures subaquàtiques. Per tant va ser la indústria la que va proporcionar l'impuls per convertir-los en vehicles rentables i fiables.

Afortunadament, des de 2005, el número de vehicles submarins autònoms AUV dedicats a la investigació científica, ha augmentat considerablement degut a diversos factors, com la miniaturització dels sistemes i la producció de vehicles més lleugers i suficientment barats com per poder ser adquirits per organitzacions no militars.

Actualment, existeix una ampla gamma de AUVs que es poden classificar segons el seu pes, mida i capacitat de càrrega d'instrumentació (*payload*) en :

- **Micro-AUV** : Petits vehicles que pesen menys de 5Kg. Desenvolupats per portar un sensor específic alhora.
- **Mini-AUV**: Aquest vehicles mesuren menys de 2 m. i tenen un pes inferior a 100 Kg. Permeten un desplegament fàcil des d'una petita embarcació, entre dues persones. Poden carregar més d'un sensor i arriben fins a profunditats de 2000m.
- **AUV-Oceanic** : Mesuren més de 2 metres i poden arribar a pesar varies tones. La càrrega útil pot excedir els 100 Kg. Poden assolir profunditats des de 600m, (Maridian 600) a 3000 m (HUGIN 3000) inclús 6000 m. (REMUS 6000). Són vehicles grans, de logística complicada. L'autonomia pot ser major a 48 hores i es necessita un vaixell mig/gran com a suport. Donada la capacitat de càrrega útil, aquest vehicles poden portar un ampli espectre de sensors i dispositius d'alt rendiment com: sonars (escombrat lateral, batimètrics, etc.), perfiladors sísmics (paramètric o 3,5 kHz), sensors CTD (Conductivitat, Temperatura, Profunditat), videocàmeres, sensors de navegació, etc.
- **Planeadors (Gliders)**: Tot i que tenen la capacitat de càrrega i la navegació limitada, es poden considerar també com AUV, ja que poden assolir trajectòries de navegació de llarga duració, recopilant dades oceanogràfiques. Només necessiten suport extern per fer el desplegament i la recuperació del vehicle. El camp d'aplicació és limitat, degut als requeriments operacionals i de *payload*.

El camp dels vehicles submarins no tripulats, es troba des de fa alguns anys en constant evolució, i té un gran potencial d'aplicació en sectors com la defensa militar, logística, fins comercials i sobretot investigació. Gràcies a les seves característiques principals, les seves reduïdes dimensions i petit pes respecte als convencionals, ofereixen una gran capacitat per adaptar-se al medi i accedir a entorns que d'una altra manera seria impossible explorar. Alhora, un factor clau pel qual moltes institucions estan adquirint i desenvolupant aquest vehicles, és la seva capacitat com a plataforma de mostreig.

Per altre banda, des de 2006 els investigadors han estat explorant l'aplicabilitat de l'ús dels AUVs per recollir dades crítiques de batimetria i caracterització del fons marí. Com és una tecnologia incipient, es requereix una gran quantitat d'investigacions i desenvolupament abans que aquest petits vehicles puguin convertir-se en eines estàndards d'hidrografia. Però existeix l'esperança de ser capaços de produir millors mapes en menys temps i menor cost utilitzant AUVs en lloc de tècniques d'estudi convencionals.

Entre els centres d'investigació que contenen avui en dia amb AUVs trobem la Unitat de Tecnologia Marina (UTM), unitat adscrita al Centre Mediterrani d'Investigacions Marines i Ambientals (CMIMA), pertanyent a l'àrea de Recursos Naturals del Consell Superior d'Investigacions Científiques (CSIC). Dintre de les activitats realitzades per aquesta Unitat trobem les de suport logístic i tècnic a les bases polars i vaixells de recerca oceanogràfica espanyols, així com d'altres relacionades amb activitats de recerca, desenvolupament i innovació en el camp de les ciències marines.

## **1.2 Objectius del projecte**

La unitat de tecnologia Marina UTM, com a unitat de servei, està contínuament buscant millors sensors, noves metodologies i plataformes per incorporar al catàleg de serveis oferts a la comunitat d'investigació marina espanyola.

Ja en 2005, va començar a buscar la viabilitat de la incorporació d'un AUV com a plataforma científica per al desplegament en vaixells petits, i a finals de 2010 UTM-CSIC va adquirir dos petits AUV.

AUV són plataformes amb gran potencial per a les operacions científiques i la UTM ha inclòs diverses línies de treball I+D relacionades amb els propis vehicles. El focus està ficat en el desenvolupament de nous sensors miniaturitzats, i la gestió i control de la qualitat de dades (processament, presentació d'informes, visualització de dades, integració de metadates etc.) amb èmfasis en l'ús d'eines de codi obert i llenguatges de programació (Python, GraSS, etc.). És en aquesta línia de recerca en la qual s'emmarca aquest projecte.

L'objectiu principal del projecte es centra en el disseny i desenvolupament d'una eina software per a la gestió i control de dades, i la generació automatitzada d'informes científics i tècnics a partir de les dades obtingudes en campanyes oceanogràfiques de vehicles autònoms submarins.

Més específicament, l'eina desenvolupada haurà de ser capaç d'oferir les següents funcionalitats:

- Lectura de fitxers .log (individualment i/o per grups) i estructuració de les dades per poder ser fàcilment bolcades en bases de dades oceanogràfiques internacionals, o per poder generar arxius en formats científics estàndards.
- Anàlisis dels principals paràmetres d'interès com la conductivitat, temperatura, profunditat etc. mitjançant visualitzacions gràfiques de les dades.
- Generació d'estadístiques de missions.
- Generació automatitzada d'informes tècnics i científics amb tota la informació d'interès.
- Batimetria a través de les dades de DVL: A partir de les dades de profunditat obtingudes pels diferents feixos del DVL construir una batimetria senzilla del terreny per on ha passat l'AUV.
- Interfície gràfica amigable, fàcil i intuïtiva, per poder ser utilitzada per qualsevol personal científic sense coneixements previs de l'aplicació
- Escalabilitat: l'eina desenvolupada ha de permetre un fàcil manteniment i una posterior ampliació o modificació.
- Generació de documentació detallada de totes les funcions i programes desenvolupats per facilitar la tasca a futurs desenvolupadors.
- Transparència en el tractament de les dades.

A més a més de les especificacions i objectius esmentats, cal afegir que el projecte s'ha desenvolupat utilitzant únicament eines de software lliure, seguint la filosofia impulsada per la UTM en els darrers anys, i que per tant el projecte constitueix per si mateix una eina de software lliure.

Com es comentarà més endavant, a l'hora d'escollir un llenguatge de desenvolupament, s'ha optat pel llenguatge Python, un llenguatge en expansió, intuïtiu i estructurat per mòduls. La utilització d'aquest llenguatge ha suposat el seu aprenentatge en paral·lel amb el desenvolupament software.

### **1.3 Interès científic de l'aplicació**

Un número creixent d'usuaris dintre de la comunitat científica, comença a adonar-se dels grans beneficis que implica l'ús de AUVs en els estudis

oceanogràfics. Per tant, són molts els esforços que s'estan portant a terme en diferents institucions arreu del món, per avançar en el desenvolupament dels AUVs i qualsevol projecte relacionat amb aquests vehicles, és d'interès per a la comunitat científica.

La informació oceanogràfica és de gran importància per un ampli rang de dominis d'aplicació i la sinergia humana i de recursos ha sigut un element clau per al creixement d'aquesta disciplina. Però actualment existeix una problemàtica, deguda a que la informació resultant de les campanyes d'adquisició de dades, utilitza paràmetres i formats diferents depenent de la naturalesa de les dades. Per tant, un dels reptes amb què s'enfronta la investigació oceanogràfica, és el d'establir un sistema adequat d'emmagatzematge i gestió de dades.

Amb l'eina de software desenvolupada en aquest projecte, es pretén crear l'estructura necessària per generar un format comú per al posterior ingrés en una base de dades.

Com a línia futura a seguir per assegurar la interoperabilitat, queda oberta la possibilitat de conversió de les dades al format NETCDF (network Common Data Format) àmpliament acceptat i utilitzat per part de la comunitat internacional per l'intercanvi de dades científiques. També ajuda a la interoperabilitat el haver adoptat l'idioma anglès per a la terminologia de paràmetres, descriptors etc. ja que és la llengua més estesa a l'àmbit tecnològic.

Per altre banda, la informació s'organitza en conjunts de dades enormes que fa que l'accés als blocs d'informació requerits en cada moment, no sigui trivial. Aquesta eina realitza un filtratge de la informació en base a variables, podent visualitzar les variables seleccionades a través de gràfiques, o estadístiques.

Resumint, el software desenvolupat:

- Facilita la cerca de les dades y metadades als arxius existents.
- Realitza una recopilació, control de qualitat i generació de reports de manera que la informació queda adequadament arxivada, documentada i disponible per a un futur.
- Facilita la distribució de les dades a diferents usuaris de diverses àrees de coneixement i requisits.
- Permet una difusió i gestió de la informació de forma clara i organitzada, on les dades d'observació es publicaran en format PDF.

A banda de tot això, que únicament és d'utilització de tipus científic, l'eina de software, serveix per contribuir al desenvolupament tecnològic, tant des del punt de vista dels sensors com del tipus d'operacions dintre de l'àmbit de UTM.

Permet portar un control de l'estat del vehicle detectant possibles falles en el funcionament, i contribueix a realitzar una bona planificació de futures campanyes, la qual cosa, degut al context socioeconòmic en què ens trobem immersos actualment, és de vital importància.

Actualment, existeixen diferents programes capaços de treballar amb dades oceanogràfiques però la gran majoria en forma de software propietari i subjecte al pagament de llicències i als criteris de desenvolupament dels seus fabricants. Aquest software, és poc adaptable a les necessitats de l'usuari i bàsicament està pensat com a eina de laboratori. En el nostre cas, el software desenvolupat podrà ser utilitzat lliurement per la comunitat científica i pot ser el punt de partida d'altres projectes, respectant les condicions de la llicència de distribució. L'extensibilitat de l'eina desenvolupada, permetrà a altres institucions científiques adaptar-la als seus vehicles, o implementar noves funcionalitats.

## 1.4 Software oceanogràfic

En investigació oceanogràfica, pel processat i anàlisi de les dades, sovint s'utilitza el software que proporciona el fabricant de la instrumentació o software propietari comercialitzat per companyies altament especialitzades en el sector. Els més utilitzats són programes com Surfer o Fledermaus que permeten la representació de les dades en 3D, originant mapes, gràfiques o imatges d'alta qualitat.

A continuació es fa una descripció de les característiques generals d'aquests paquets de software:

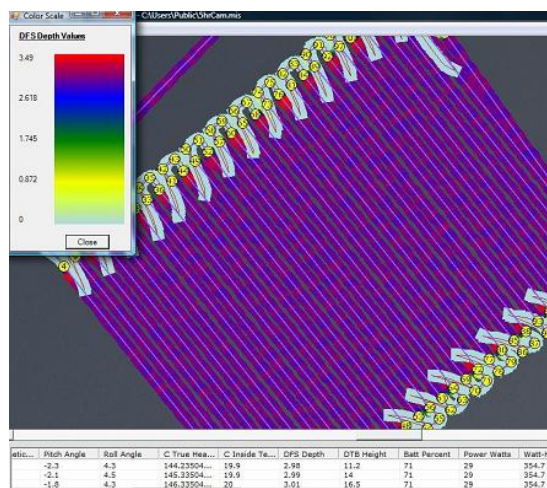
### VectorMap

Normalment el fabricant del vehicle proporciona un software específic pel tractament de les dades. En aquest cas, tots els models d'AUV Iver2 d'OceanServer, venen amb un software de sèrie anomenat VectorMap.

VectorMap (VM) és una aplicació de software per a Windows, que es fa servir per a la planificació de missions i com a eina de registre. La compatibilitat de VectorMap amb les cartes nàutiques de NOAA ENC<sup>®</sup> (NOAA Electronic Navigational Charts) permet al operador desenvolupar intuïtivament la planificació de la missió fent un simple clic sobre el mapa. També s'encarrega de registrar i generar arxius de dades .log fàcils d'exportar a altres eines software d'anàlisi de dades.

A banda d'això, el software del fabricant compta amb una eina de post-processat de les dades. Després de completar la missió, permet descarregar les dades registrades pel vehicle i els sensors i superposar-les. Per tant és pot utilitzar com a eina de presentació de dades, i graficar les diferents variables per a la validació y depuració de les dades. VM pot crear un mapa de colors de qualsevol conjunt de dades de l'arxiu .log. (Fig. 1.2)

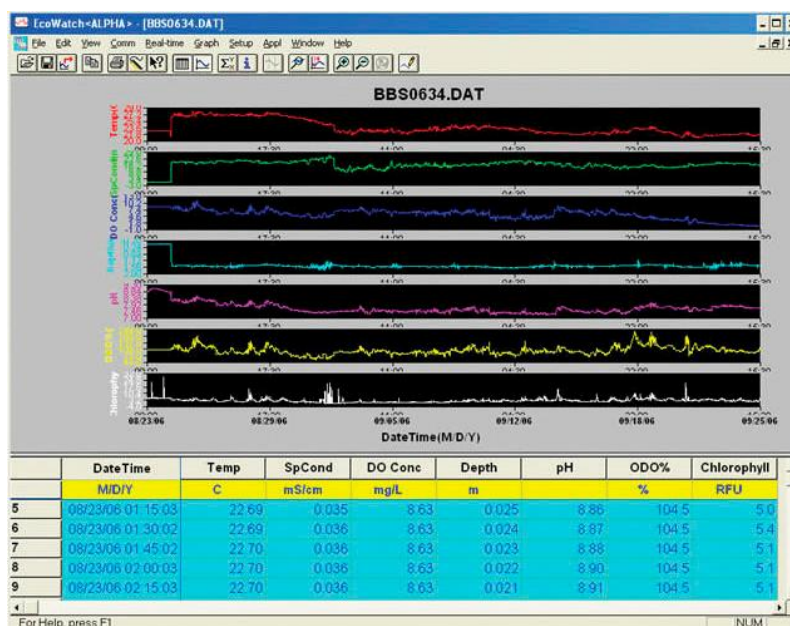




**Fig. 1.2** Mapa de color de la profunditat sobre la trajectòria amb VectorMap

## EcoWatch

És un software propietat de la companyia YSI Inc. YSI és dedica principalment a desenvolupar i fabricar sensors, instrumentació, software i plataformes per a la recol·lecció i anàlisi de dades. EcoWatch consisteix en un software simple d'anàlisi de dades per a les sondes *YSI 6-Series Sondes*. Permet realitzar una visualització de les dades mitjançant representacions gràfiques bàsiques. Té una interfície simple i senzilla i pot executar estudis de conjunts de dades superiors a 150000 mostres.



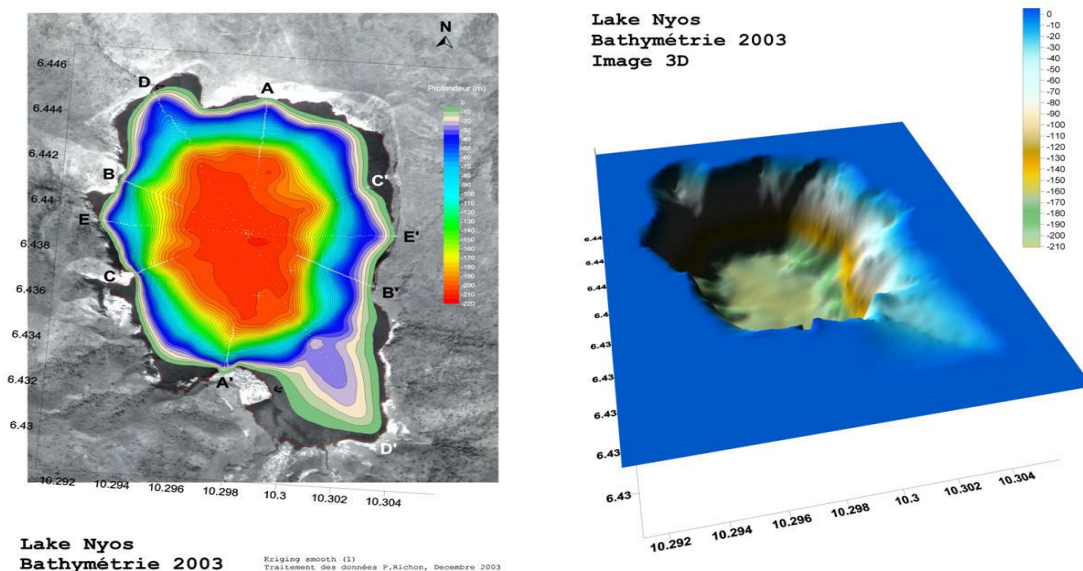
**Fig. 1.3** Interfície EcoWatch: Representació de les variables dels sensors.

## Surfer

És un software de la companya Golden Software molt complet per a la visualització en 3D, la creació d'isolínies i el modelat de superfícies que s'executa sota Windows. S'utilitza àmpliament pel modelatge 3D del terreny, per batimetries (Fig. 1.4), anàlisis de superfícies, corbes de nivell, volumetria etc. És un dels softwares més utilitzats tant per oceanògrafs com geòlegs, arqueòlegs, geofísics etc.

Surfer és un programa especialitzat en *gridding* que és el procés mitjançant el qual s'utilitzen els valors dispersos de les dades per crear una hipotètica malla o *grid* de valors en tots els punts d'una matriu rectangular. Per realitzar mapes de contorn o gràfiques de superfície és requereix que els punts estiguin distribuïts regularment en els arxius de *grid* amb extensió [.GRD]. Com el conjunt de dades XYZ originals és irregular (espaïats de manera irregular), existeixen "forats" en la distribució dels punts. Mitjançant el *gridding* s'omplen aquest forats per extrapolació o interpolació dels valors de Z, produint un array de Z amb valors regularment espaïats. Aquests valors del grid són els que s'esperarien si les mesures s'haguessin pres en tots els punts de la matriu. Surfer compta amb un sofisticat motor d'interpolació que permet escollir entre els següents algorismes: *Inverse Distance*, *Kriging*, *Minimum Curvature*, *Polynomial Regression*, *Triangulation*, *Nearest Neighbor*, *Shepard's Method*, *Radial Basis Functions*, *Natural Neighbor*, *Moving Average*, y *Local Polynomial*.

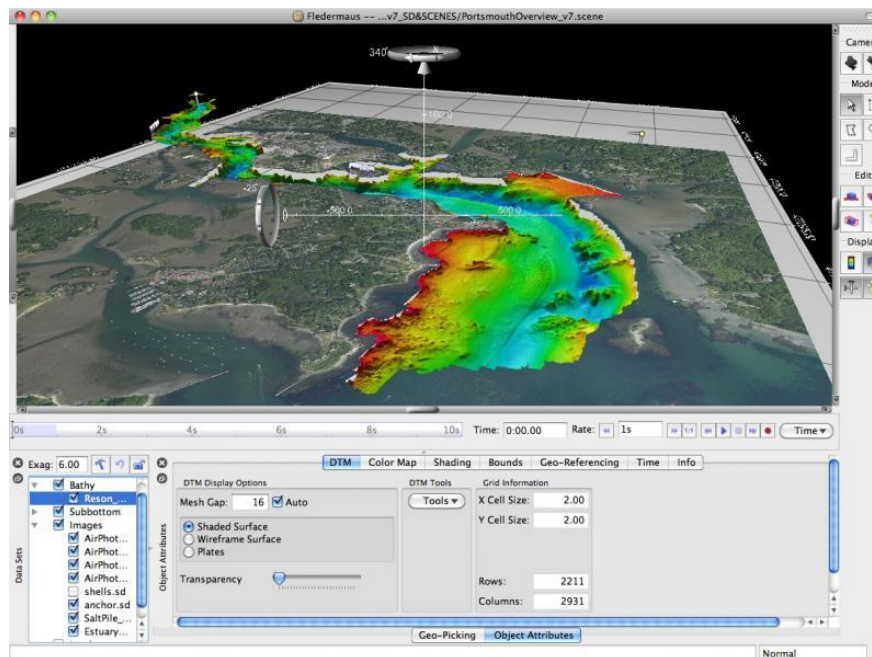
També permet aplicar càlculs sobre les dades, ja que compta amb algunes funcions matemàtiques (trigonomètriques, exponencials i de Bessel) i funcions estadístiques aplicables sobre columnes.



**Fig. 1.4** Línies de contorn de profunditat i representació en 3D mitjançant el software Surfer per Patrick Richon.

## Fledermaus

És un software molt potent de la companyia QPS (Quality Positioning Services BV) utilitzat per a l'anàlisi i visualització 3D de dades topogràfiques i batimètriques. És pot executar tant en Windows com Linux, permet analitzar grans conjunts de dades i accepta diferents formats per importar les dades a l'escena de 3D. Fledermaus també permet visualitzar en temps real les dades provinents de ROVs, vaixells o altres plataformes. Les amplies funcionalitats de Fledermaus permeten fer-lo servir en les diferents fases d'un projecte, des de la planificació, processat i control de qualitat, fins a l'anàlisi i producció d'imatges, gràfiques i animacions.



**Fig. 1.5** Exemple de batimetria generada amb Fledermaus

## Ocean Data View (ODV)

Una altre alternativa de software amb un gran nombre d'usuaris en oceanografia és Ocean Data View (ODV), un software propietari desenvolupat en el centre *Alfred Wegener Institute for Polar and Marine Research*. Aquest software és pot utilitzar de manera gratuïta sempre i quan sigui per fins d'investigació o ensenyança no comercials ni militars. Actualment és troba disponible la versió 4 (ODV4).

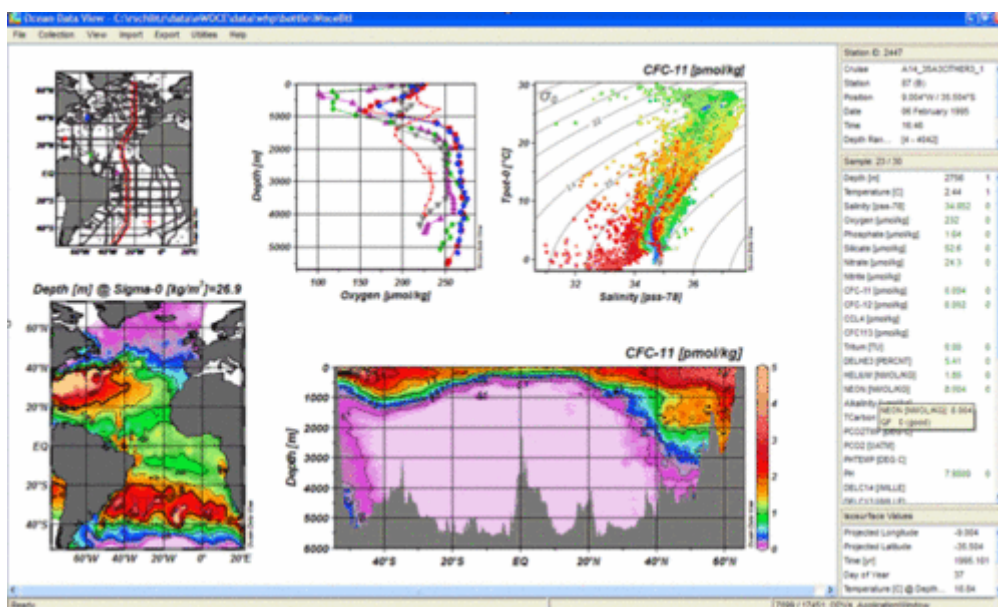
ODV és un paquet de software per a l'anàlisi i visualització de dades oceanogràfiques i altres dades geo-referenciades. S'executa sota Windows, Mac OS X, Linux i Unix.

És especialment útil per produir:

- Gràfiques de les propietats de les estacions seleccionades

- Diagrames de dispersió per grups d'estacions
- Seccions verticals a color
- Distribucions de color en iso-superfícies
- Gràfiques de l'evolució temporal
- Animacions

ODV pot mostrar els punts de les dades originals o realitzar una malla (grid) basada en les dades originals. Per això, ODV compta amb dos algoritmes de promig ràpid i compta amb un software (DIVA) integrat per a la interpolació avançada. Suporta diferents projeccions cartogràfiques, el que permet produir mapes d'alta qualitat i exportar els resultats a múltiples opcions com PostScript, gif, png o jpg. El format de les dades ODV permet l'emmagatzemament de grans bases de dades i un accés ràpid a aquestes. Les dades de projectes com ARGO, WOCE, World Ocean Database, World Ocean Atlas, SeaDataNet, Medar/Medatlas es poden importar directament en ODV i també és compatible amb el format netCDF



**Fig. 1.6** Interfície Ocean Data View

Després de l'exposició dels diferents paquets de software existents, es pot observar que les funcionalitats requerides per aquest projecte no es troben en el software comercial, ja que pertanyen a un àmbit concret i han de satisfer els requeriments específics de l'UTM. El software del fabricant, VectorMap o EcoWatch és molt reduït en quant a prestacions i sofisticació. Només permet realitzar un tipus de gràfica senzilla i exclusivament de les variables dels sensors. El software desenvolupat en aquest projecte, a banda de les variables físico-químiques recollides pels sensors, també permet representar totes les variables d'enginyeria, necessàries per portar un control sobre l'estat del vehicle i dels sensors.



Fledermaus i Surfer focalitzen el seu objectiu en realitzar gràfiques en 3D d'alta qualitat d'un conjunt de dades, oferint també en el cas de Surfer la possibilitat de calcular estadístiques bàsiques de les columnes de dades.

El software desenvolupat no es pot comparar amb aquestes dues eines en quant a prestacions gràfiques, ja que aquest no implementa, (en aquesta primera versió) algoritmes d'interpolació per a la generació de gràfiques en 3D. Per tant les possibilitats de generar mapes batimètrics de qualitat similar als que ofereixen Fledermaus o Surfer, encara està molt lluny d'aquesta primera versió que només s'encarrega d'obtenir les coordenades batimètriques, exportar-les a un fitxer ASCII i representar els punts obtinguts en 2 D. Tot i que Surfer també permet realitzar estadístiques bàsiques, com màxims, mínims etc., aquestes només s'apliquen sobre les columnes de dades de les diferents variables individualment. El software desenvolupat compta amb un mòdul independent pel càlcul d'estadístiques bàsiques, que implementa els algoritmes necessaris per generar resultats de diferents variables alhora i de múltiples fitxers.

Per últim dir, que el software Ocean Data View, és el que més es sembla en quant a objectius i prestacions. El software desenvolupat es centra en l'àmbit d'aplicació dels vehicles autònoms submarins, mentre que ODV permet un anàlisi interactiu de diferents estacions i plataformes. A més el tipus de format de dades de ODV permet l'accés a diferents bases de dades oceanogràfiques i és compatible amb el format netCDF, encara no implementat en el software desenvolupat en aquest projecte.

## CAPÍTOL 2. ENTORN DE DESENVOLUPAMENT

Aquest capítol descriu l'entorn de desenvolupament del projecte i altres aspectes que s'han cregut necessaris per facilitar la posada en situació del lector. Els diferents apartats descriuen els elements implicats en el projecte (tant hardware com software), les seves característiques, com es pretenen utilitzar i amb quina finalitat.

### 2.1 Característiques dels vehicles

Com ja s'ha comentat anteriorment, a finals de 2010 UTM-CSIC va adquirir dos petits AUV OceanServer Iver2-580-EP35. El desenvolupament de l'eina de software d'aquest projecte s'ha basat en aquest models, tot i que el disseny de l'eina en qüestió s'ha flexibilitzat per poder donar suport a altres models de vehicles.

Els seus paràmetres de funcionament han estat necessaris a l'hora de desenvolupar l'aplicació. Per aquest motiu, es creu convenient fer una breu descripció de les seves característiques.



**Fig. 2.1** Vehicles mini AUV configuració EcoMapper.

Els vehicles estan basats en el model OceanServer Iver2-580-EP35 amb les següents especificacions comuns:

- **Forma:** Tubular – torpedo.
- **Mida:** 140 x 15 cm
- **Pes** (en aire): 21 kg + sensors
- **Profunditat màxima d'operació:** 100 m (en configuració estàndard).
- **Autonomia:** 8+ hrs (600 W/h batt) @ 2 nusos.
- **Comunicacions:** WiFi.
- **Navegació:** GPS, DVL.
- **Processadors:** 2 CPU's basades en processador Àtom (un dedicat a futures expansions amb 8 ports USB y 16 ports sèrie), incloent API per interfície amb el processador principal.

- **Memòria:** 2 Discs durs d'estat sòlid ( 60Gb)
- **Ports:** 3 Ports accessibles en el casc per instal·lar la instrumentació addicional i espai interior per 3 plaques PC-104.
- **Operativitat:** Llançament i recuperació per un sol home des de petites embarcacions o des de la costa.
- **Seguretat:** Airbag de seguretat

Els dos vehicles disposen de capacitat per incorporar sensors addicionals i permeten dues configuracions:

- **Configuració Ecomapper:** Permet el control de la qualitat del aigua realitzant un mostreig d'alta resolució en aigües someres (litorals i interiors). Està equipat amb un ADCP/DVL Sontek d'1MHz (up/down) per fins de navegació i perfils de corrents. També conta amb una sonda multiparamètrica YSI 6600 amb sensors de conductivitat, profunditat, temperatura, oxigen dissolt, clorofil·la, pH, turbidessa.
- **Configuració Imatge:** Aquest vehicle està destinat a la integració i prova de nous desenvolupaments, així com la realització de missions amb càmeres i *sidescan* sonar. Està equipat amb un ADCP/DVL (4feixos/1altímetre), *sidescan* sonar de doble freqüència (330/800 KHz) i càmeres de baixa resolució.

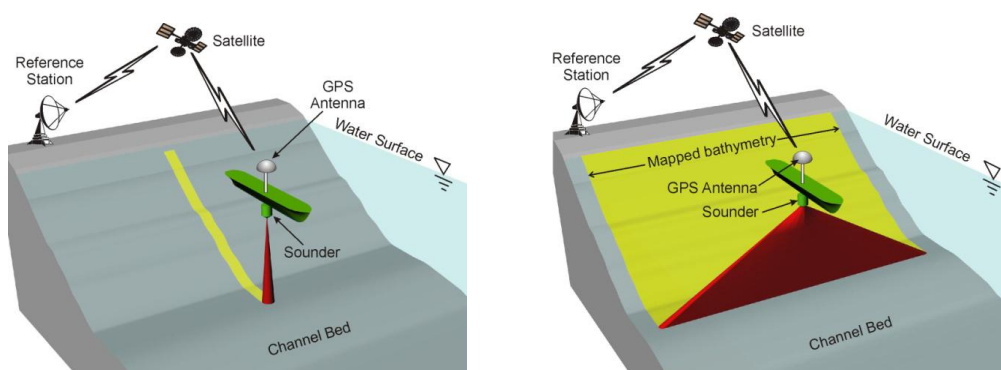
Amb la configuració ecomapper, aquest model de vehicle, genera un únic arxiu de sortida .log on s'emmagatzemen totes les dades recollides pels sensors i variables d'estat del vehicle. Aquest arxiu és el que s'utilitzarà com a font de dades en el software desenvolupat.

## 2.2 Batimetria i ADCP

Nombrosos sectors, tant comercials com d'investigació requereixen dades geofísiques d'alta resolució i comprensibles. Avui en dia resulta imprescindible contar amb aquestes dades per tractar aspectes com la seguretat en la navegació marítima, l'exploració i explotació dels recursos marins, així com la preservació i la defensa d'aquest ambient.

La finalitat doncs d'una batimetria, és l'obtenció de les coordenades XYZ del punts submergits, i d'aquesta manera poder realitzar mapes que permeten caracteritzar i conèixer el fons marí.

Un desenvolupament relativament nou en la obtenció de dades batimètriques, és l'ús combinat d'eco sondes amb el sistema de posicionament global GPS, obtenint, d'aquesta manera, una sèrie de punts (x,y,z) per a la representació tridimensional de la morfologia del fons.



**Fig. 2.2** Sistemes batimètrics; ecosondes monofeix i multifeix + GPS

Existeix una forma alternativa per obtenir la batimetria del fons, i es mitjançant l'ús d'un correntòmetre d'efecte Doppler (ADCP).

### 2.2.2 AUV/DVL-ADCP

Normalment els AUV estan equipats amb un DVL/ADCP. Un ADCP (Acoustic Doppler Current Profile) és un perfilador acústic de corrents, el què vol dir que dóna les components de la velocitat del aigua en diferents capes de la columna d'aigua. El sistema consta d'un transductor que emet ones acústiques i una unitat electrònica que genera els polsos i pre-processa les ones rebudes. El seu principi de funcionament es basa en l'efecte Doppler, transmeten so a una freqüència fixa i escoltant els ecos retornats pels reflectors de l'aigua. Aquests reflectors són petites partícules o plàncton que reflexen el so cap a l'ADCP. Es troben flotant i es mouen a la mateixa velocitat horitzontal que l'aigua. Quan el so enviat per l'ADCP arriba als reflectors, aquest es troba desplaçat a una major freqüència degut a l'efecte Doppler. Aquest desplaçament freqüencial es proporcional a la velocitat relativa entre el ADCP i els reflectors. Part d'aquest so desplaçat freqüencialment es reflexa cap a l'ADCP on es rep desplaçat per una segona vegada. La fórmula que relaciona la velocitat amb la freqüència es:

$$F_d = 2F_s (v/c) \quad (2.1)$$

on  $F_d$  és el desplaçament Doppler en freqüència.

L'ADCP/DVL incorporat a l'AUV d'estudi, consta de 4 transductors d'1MHz i feixos orientats  $25^\circ$  respecte a la vertical. Si el vehicle es troba suficientment a prop del fons marí, el ADCP incorporat pot captar les reflexions del fons i determinar la profunditat dels punts d'impacte dels feixos. Internament l'ADCP estima la profunditat a partir de la integració de les components de velocitat dels feixos.

Aquestes dades junt amb les d'actitud del vehicle (*heading*, *roll*, *pitch*) es poden utilitzar per mapejar les posicions XYZ de la intersecció del raig amb el terra.



En el següent apartat, es detalla el càlcul d'aquestes coordenades.

### 2.2.3 Batimetria a partir del DVL-ADCP

Les mesures de l'ADCP, adquirides en els feixos individuals, inclouen informació de dades batimètriques a través dels mesuraments de profunditat.

Per obtenir les coordenades geogràfiques de cada feix es segueix el següent procediment:

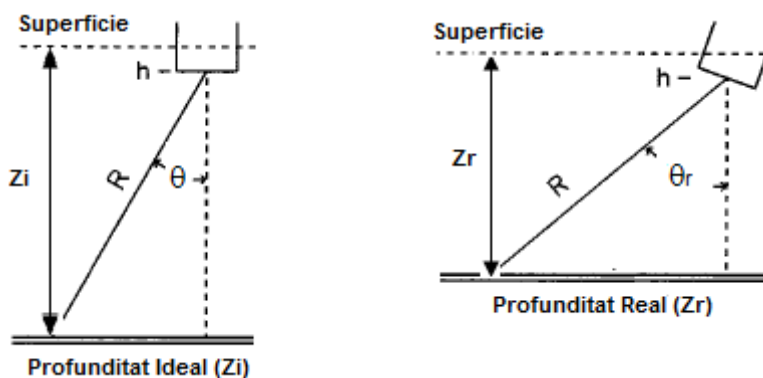
Primer, es corregeix la profunditat del feix, respecte als angles d'inclinació del vehicle (pitch/roll). Segon, es transforma la mesura de la profunditat corregida, al sistema de coordenades de la Terra i finalment, s'afegeix la informació de posicionament, latitud i longitud, per obtenir les dades batimètriques, és a dir les coordenades del fons (x,y,z). A continuació, s'exposen els càlculs detallats del procés.

Per cada feix, el software d'adquisició de dades que porta l'instrument calcula la profunditat  $Z_i$  a partir de la distància del raig  $R_i$  amb la relació:

$$Z_i = R_i \cdot \cos(\theta) + h \quad (2.2)$$

on  $\theta$  és l'angle del feix (típicament  $25^\circ$ ) i  $h$  és la profunditat a la què es troba el transductor respecte a la superfície.

Però existeix un inconvenient, i és que el software assumeix que els angles pitch i roll del transductor són 0. Quan el vehicle està girat (pitch i/o roll diferent de 0), l'angle que forma el feix respecte a la vertical ( $\theta_r$ ) no és l'ideal, és a dir, que la mesura de la profunditat que ens dona el software no és real.



**Fig. 2.3** Profunditat ideal vs. profunditat real

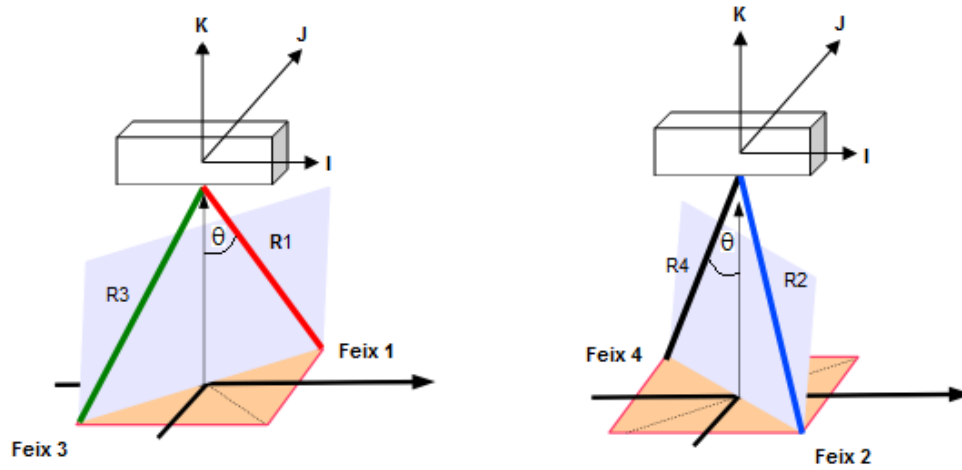
Per tant, cal corregir la sortida  $Z$ , amb respecte al pitch/roll per obtenir la profunditat real del raig. Una manera de resoldre aquest problema i obtenir la

profunditat real, implica tenir en compte l'angle actual amb respecte a la vertical ( $\theta_r$ , en funció de pitch i roll)

$$Z_r = R_i \cdot \cos(\theta_r) + h \quad (2.3)$$

Tanmateix, el càlcul de l'angle real  $\theta_r$ , no és trivial. Una alternativa és utilitzar l'algoritme que es fa servir per corregir la velocitat [1]. Aquest pot ser aplicat de manera similar per corregir la profunditat del raig de la manera següent :

La posició de contacte de cada raig amb el terra, ve donada a partir de les components (i, j, k) de R en el sistema de coordenades de l'instrument (I, J, K).



**Fig. 2.4** Posició d'intersecció i, j, k per als feixos 1-3 i 4-2

Per tant, per cada feix, les coordenades d'intersecció amb el fons (abans de les correccions de roll/pitch) són:

$$\begin{aligned} \text{Feix1} &= (\Delta I, \Delta J, \Delta K) = R_1(\sin\theta \tan 45, \sin\theta \tan 45, -\cos\theta) \\ \text{Feix2} &= (\Delta I, \Delta J, \Delta K) = R_2(\sin\theta \tan 45, -\sin\theta \tan 45, -\cos\theta) \\ \text{Feix3} &= (\Delta I, \Delta J, \Delta K) = R_3(-\sin\theta \tan 45, \sin\theta \tan 45, -\cos\theta) \\ \text{Feix4} &= (\Delta I, \Delta J, \Delta K) = R_4(-\sin\theta \tan 45, -\sin\theta \tan 45, -\cos\theta) \end{aligned} \quad (2.4)$$

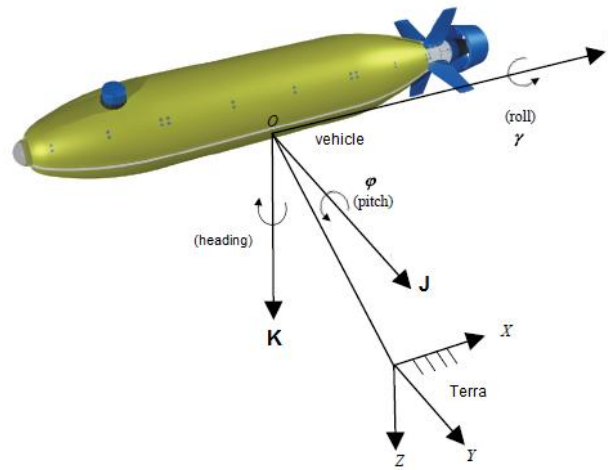
On  $R_i$  s'obté aplicant la inversa a l'equació (2.2) i  $Z_i$  és la profunditat 'ideal' registrada pel software del ADCP.  $\Delta K$  en el sistema de coordenades del feix, és sempre negatiu, ja que es defineix com a positiva la direcció cap amunt.

Coneixent l'actitud del vehicle, és a dir els angles pitch i roll, la posició d'intersecció del feix amb el fons  $(\Delta I_i, \Delta J_i, \Delta K_i)$  es transforma al sistema de coordenades del vehicle, on els angles de roll/pitch s'apliquen a cada feix a

partir de les transformacions d'Euler. Per tant, aplicant la matriu de rotació (2.5) s'aconsegueix corregir simultàniament el roll/pitch en les mesures de profunditat.

$$\begin{pmatrix} \Delta I_v \\ \Delta J_v \\ \Delta K_v \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi & \cos \varphi \end{pmatrix}}_{\text{pitch correction}} \underbrace{\begin{pmatrix} \cos \gamma & 0 & \sin \gamma \\ 0 & 1 & 0 \\ -\sin \gamma & 0 & \cos \gamma \end{pmatrix}}_{\text{roll correction}} \begin{pmatrix} \Delta I_i \\ \Delta J_i \\ \Delta K_i \end{pmatrix} \quad (2.5)$$

Es realitza un primer gir d'angle  $\gamma$  al voltant de l'eix I, i un segon gir d'angle  $\varphi$  sobre l'eix J.



**Fig. 2.5** Sistema de coordenades del vehicle

En aquest moment, es tenen els punts d'intersecció en coordenades del vehicle  $(\Delta I_v, \Delta J_v, \Delta K_v)$  així que és necessari un tercer gir d'angle  $\phi$  (heading) sobre l'eix K, per obtenir els punts d'intersecció dels feixos amb el fons, en coordenades de la Terra (2.6).

$$\begin{pmatrix} \Delta I_e \\ \Delta J_e \\ \Delta K_e \end{pmatrix} = \begin{pmatrix} \cos \phi & \sin \phi & 0 \\ -\sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \Delta I_v \\ \Delta J_v \\ \Delta K_v \end{pmatrix} \quad (2.6)$$

Finalment, per obtenir les coordenades geogràfiques en tres dimensions (latitud, longitud, profunditat) dels punts del fons:

$$X = X_0 + \Delta I_e, \quad Y = Y_0 + \Delta J_e, \quad Z = Z_0 - \Delta K_e \quad (2.7)$$

On  $X_0$  i  $Y_0$  són les coordenades UTM Est i Nord de l'ADCP i  $Z_0$  és la profunditat del transductor.

El software desenvolupat, implementa l'algoritme descrit en aquest apartat, i d'aquesta manera dota a l'aplicació de la capacitat de calcular les coordenades geogràfiques ( $X_e, Y_e, Z_e$ ) de cada feix, per al posterior aixecament batimètric.

## 2.3 Eines software i llenguatge de programació

L'avenç en la comunitat científica sempre ha anat acompanyat per la lliure circulació d'idees i coneixement. Amb l'ús de software lliure, es pretén impulsar aquesta filosofia i contribuir a generar més coneixement mitjançant processos col·laboratius, voluntaris, lliures i oberts. Per tant, és lògic pensar que tot projecte dintre de l'àmbit de la investigació hauria de ser lliure. Aquesta és la filosofia de desenvolupament que des de fa uns anys està seguint UTM.

Dit això, en aquest apartat s'introdueixen els conceptes de software lliure, llenguatge de programació Python, i les eines utilitzades.

### 2.3.1 Software lliure

Anomenem software lliure aquell que atorga als usuaris la llibertat per executar, copiar, distribuir, estudiar, canviar i millorar el software. És convenient no confondre el software lliure amb el software gratuït. Per Richard Stallman, creador del software lliure, el software lliure és una qüestió de llibertats i no de preu.

De manera més precisa, es refereix a quatre llibertats dels usuaris del software:

- Llibertat 0 : La llibertat d'utilitzar el programa, amb qualsevol propòsit.
- Llibertat 1: La llibertat d'estudiar com funciona el programa i adaptar-lo a les pròpies necessitats. El accés al codi Font és una condició prèvia per això.
- Llibertat 2: La llibertat de distribuir còpies, amb les que pots ajudar a tercers.
- Llibertat 3: La llibertat de millorar el programa i fer públiques aquestes millores a la resta, de manera que tota la comunitat es beneficiï.

Per tant, la distribució del software ha d'anar acompanyat del codi font (codi obert o *open source*) per poder fer efectives les llibertats que el caracteritzen.

Dintre del software lliure existeixen matisos a tenir en compte. Per exemple, el software lliure de domini públic, significa que no està protegit per *copyright*, per tant es podrien generar versions no públiques d'ell. En canvi, el software lliure

protegit per *copyleft*, garanteix que les modificacions continuaran sent lliures. És a dir que com en qualsevol software propietari, el software lliure es distribueix sota diferents llicències que plasmen els drets i deures d'usuaris i autors així com les condicions d'ús.

Existeix un gran nombre de llicències per a la distribució de software lliure però la més utilitzada és la GNU General Public License (GPL), sota la qual es pensa distribuir el software desenvolupat en aquest projecte. És un tipus de llicència per a programari que permet la còpia, distribució i modificació del software, sempre que qualsevol derivació d'aquest es continuï distribuint amb la mateixa llicència GPL. D'aquesta manera s'assegura que les modificacions del software continuaran sent lliures.

### 2.3.2 Llenguatge Python

En els últims temps, Python està guanyant molts adeptes entre la comunitat de desenvolupadors ja que es tracta d'un llenguatge fàcil d'aprendre, a l'hora que potent. Disposa d'eficients estructures de dades d'alt nivell, i una senzilla, tot i que efectiva, aproximació a la programació orientada a objectes. La sintaxi elegant i fàcil d'escriure de Python, juntament amb la seva naturalesa de llenguatge interpretat, el fan ideal per la construcció de scripts i el ràpid desenvolupament d'aplicacions en diferents àrees i sobre la majoria de les plataformes.

Més específicament es pot dir que Python és un llenguatge interpretat, multiplataforma i multiparadigma, és a dir, que permet diversos estils com la programació orientada a objectes, programació estructurada i programació funcional. A més a més, es tracta d'un llenguatge modular i extensible que permet dividir el programa en mòduls reutilitzables des d'altres programes Python. Aquesta característica el fa idoni pel desenvolupament d'aquest projecte, ja que es pretén que l'aplicació resultant pugui adaptar-se a diferents necessitats i ser extensible en un futur.

Altres característiques que decanten la balança a favor de l'ús de Python en aquest projecte, és que és un llenguatge lliure i obert i conta amb nombroses biblioteques estàndard que ofereixen ajuda en diferents aspectes com: expressions regulars, generació de documentació, depuració d'unitat, la creació d'interfícies gràfiques, etc.

### 2.3.3 Eines software de desenvolupament

En el desenvolupament de l'aplicació s'han utilitzat diverses eines de software lliure, per tant ha sigut necessària la instal·lació d'un editor de llenguatge Python i de llibreries específiques com a suport en la implementació dels diferents mòduls. Després d'un estudi comparatiu entre diverses opcions disponibles, s'han escollit les següents llibreries:

- **Editor Python:** Spyder
- **Llibreries Python:**
  - ReportLab
  - Matplotlib
  - Numpy
  - PyQt4

**ReportLab** és una llibreria per crear documents complexes mitjançant programació en Python. Tot i que el conjunt d'eines de ReportLab a evolucionat al llarg dels anys, no totes les llibreries amb què disposa són open source.

Conta amb un motor de disseny de pàgines flexible anomenat Platypus que permet dissenyar pàgines i topografies fent servir scripts. Això facilita la tasca d'incloure gràfics, capçaleres, paràgrafs o taules.

Entre totes les possibles solucions per generar pdf en Python s'ha escollit Reportlab, per la quantitat de possibilitats que ofereix alhora de crear un document amb una mínima qualitat.

**Matplotlib** és una llibreria per a la generació de gràfiques en 2D/3D a partir de les dades contingudes en llistes o arrays en el llenguatge de programació Python i la seva extensió matemàtica Numpy. Proporciona una API, pylab, dissenyada per recordar a la de MATLAB.

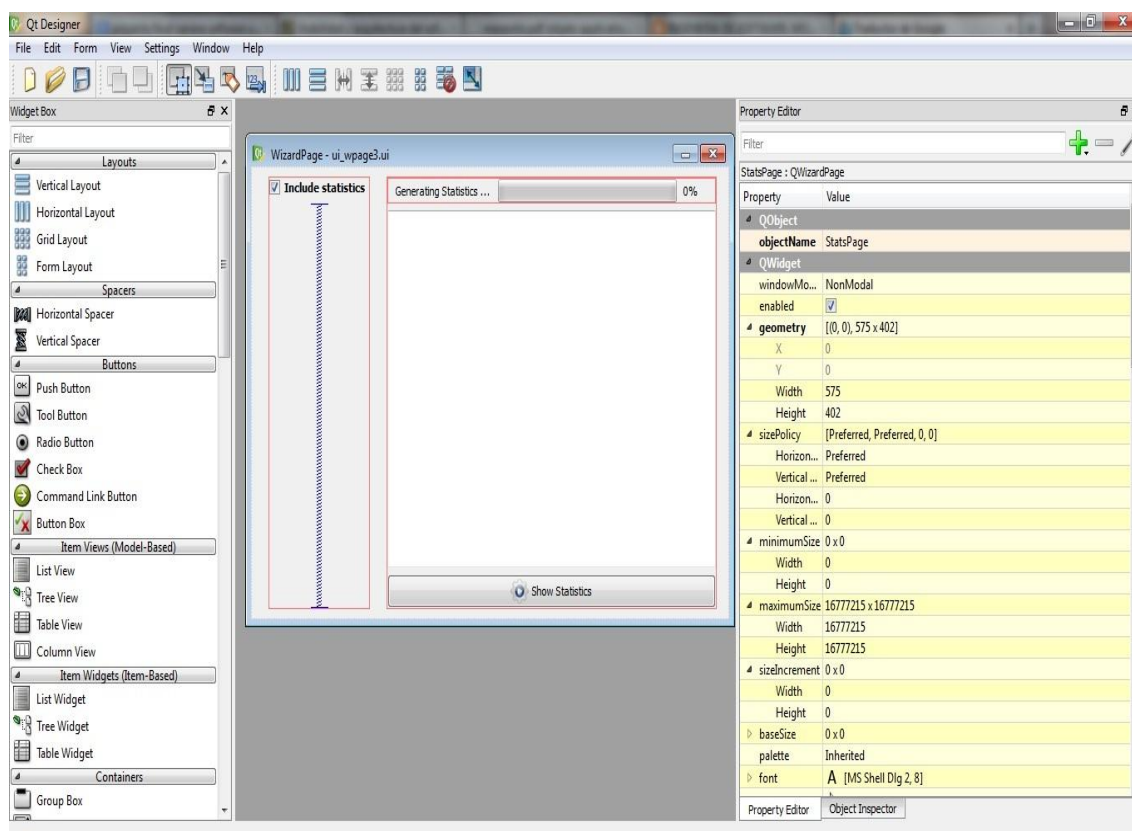
**Numpy** és una extensió de Python que li aporta major suport pel tractament de vectors i matrius. Constitueix doncs, una biblioteca de funcions matemàtiques d'alt nivell per operar amb aquest vectors o matrius.

I per últim, cal parlar de **PyQt**. Qt és una biblioteca multiplataforma, dissenyada per desenvolupar interfícies de manera còmoda i senzilla. PyQt és la implementació de Qt amb llenguatge Python . Està desenvolupada per la signatura britànica Riverbank Computing i disponible per a Windows, GNU/Linux i MAC OS X sota diferents llicències (GNU v2 i GNU v3) per al desenvolupament de programari de codi obert (*open source*) i programari lliure, i una altra de pagament per al desenvolupament d'aplicacions amb qualsevol llicència. Existeixen diferents projectes que fan servir Qt, com per exemple Google Earth, Skype o KDE entre altres.

Per al desenvolupament d'aquest projecte s'han testejat altres biblioteques gràfiques com PyGTK o WxPython però finalment s'ha escollit PyQt. Les facilitats que ofereix alhora de crear interfícies, la seva flexibilitat, la propietat de ser multiplataforma i una eina de codi obert fa PyQt especialment interessant per al context d'aquest projecte.

A més, Qt inclou Qt Designer, un dissenyador d'interfícies gràfiques per facilitar el seu desenvolupament. D'aquesta manera permet separar la lògica de la part

purament visual. En la figura següent (Fig. 2.6) es mostra una imatge d'aquesta eina amb el disseny d'un dels formularis de AUVsoft.



**Fig. 2.6** Qt Designer, exemple del procés de disseny d'un formulari

## **CAPÍTOL 3. DISSENY SOFTWARE**

### **3.1 Desenvolupament del software**

Tractant-se d'una eina de software lliure, aquest projecte requereix un disseny clar i fàcil d'entendre per propers desenvolupadors que vulguin utilitzar-lo o estendre'l per adaptar-lo a les seves necessitats.

Per tant, es necessari seguir un procés de desenvolupament per tal de segmentar el projecte en diferents parts i assegurar la seva qualitat i reutilització futura. El procés de desenvolupament de software, potser descrit generalment com una seqüència d'activitats que cal dur a terme per arribar al producte final.

Específicament, s'ha dividit aquest procés en les diferents parts:

#### **Anàlisis dels requeriments**

Aquesta és la primera etapa en qualsevol projecte de desenvolupament de sistemes o de software. L'objectiu principal és identificar les necessitats de l'usuari i definir les condicions que el producte ha de tenir per assolir els objectius. Alhora de definir els requeriments, els podem classificar en dos tipus:

- Funcionals: Descriuen els serveis que el software ha de ser capaç de proporcionar, sense definir detalls d'implementació. Donen una idea precisa del problema que s'ha de solucionar. Especifiquen el tipus d'entrada/sortida.
- No funcionals: Defineixen propietats o restriccions del sistema com pot ser el rendiment, fiabilitat o portabilitat entre d'altres.

#### **Arquitectura del software**

L'arquitectura del software correspon al nivell conceptual més alt. Descriu l'organització fonamental del sistema, és a dir la seva estructura, definint els components i les relacions entre ells. Existeixen patrons coneguts, que es poden adoptar segons les necessitats, com el model client - servidor o l'arquitectura de tres nivells.

#### **Disseny**

És l'etapa de disseny de baix nivell. Es realitza una descripció precisa i detallada de cada subconjunt de l'aplicació.

Es defineixen especificacions de la codificació i la tecnologia que s'utilitzarà en la implementació final.





## 3.2 Anàlisi dels requeriments

Per satisfer les necessitats dels diferents usuaris del software, es fa un anàlisi previ dels diferents casos d'ús del software i de les funcionalitats que s'espera que tinguin. En aquest cas, els requeriments que s'espera que tingui el software, venen marcats per les necessitats d'UTM, que ja s'han descrit prèviament, com els objectius a assolir per aquest projecte. Tanmateix, aquest pas ha sigut necessari en el disseny del software, ja que inicialment es contava amb unes especificacions a grans trets del que es volia que fes el software. Recordant els objectius, és vol que el software sigui capaç de:

Funcionals:

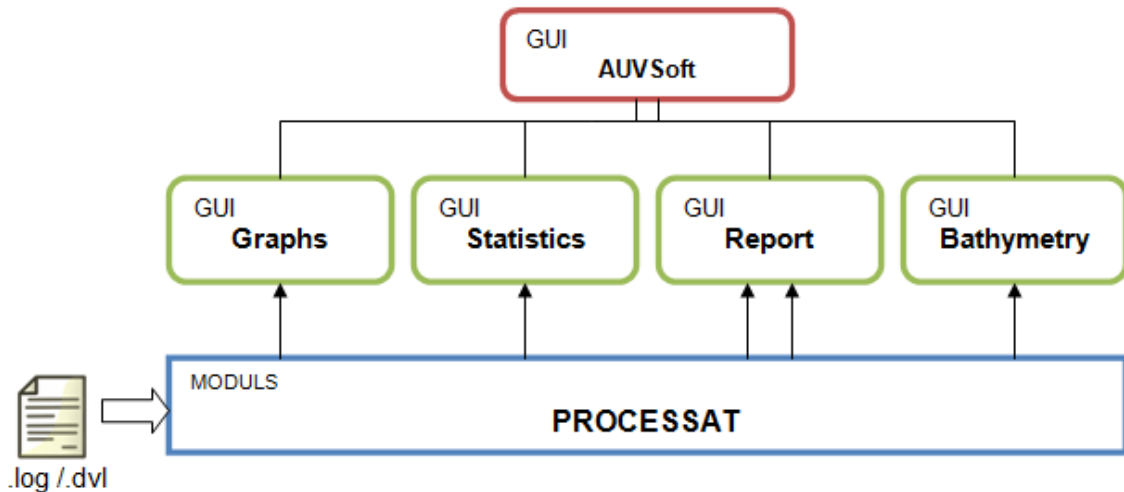
- Lectura de fitxers .log (individualment i/o per grups).
- Estructuració de les dades per poder ser fàcilment bolcades en BBDD o poder generar arxius en formats científics com NETCDF.
- Generació automatitzada d'informes científics o tècnics en format pdf a partir dels arxius d'estat seleccionats. Aquest informes han de contenir la informació referent al projecte o campanya al qual pertanyen les dades analitzades, estadístiques de missions i representacions gràfiques dels diferents paràmetres d'interès.
- Batimetria del terreny, a partir dels feixos del DVL/ADCP.
- Generació d'estadístiques de missions.
- Visualització gràfica de les diferents variables obtingudes amb l'AUV.

No funcionals:

- Escalabilitat: El sistema s'ha de construir sobre la base d'un desenvolupament evolutiu e incremental. Ha de permetre modificar o incloure noves funcionalitats sense afectar el codi existent
- Facilitat d'ús: Per garantir una interacció senzilla amb el sistema, totes les funcionalitats implementades han de ser accessibles mitjançant una interfície gràfica. Aquesta GUI ha de tenir un funcionament senzill, amigable i intuïtiu per poder ser utilitzat per personal científic sense coneixements previs de l'aplicació
- Manteniment: Tot el codi font ha d'estar completament documentat.
- Llenguatge de programació: Python

### 3.3 Arquitectura del software

En aquesta secció, s'explica l'arquitectura final del software, basada en els requeriments i funcionalitats prèviament descrits i en l'experiència obtinguda en el desenvolupament dels primers prototips.



**Fig. 3.2** Estructura general del software

Una de les estructures més conegudes alhora de dissenyar una arquitectura és la de tres capes. Es tracta d'una arquitectura amb un gran avantatge, i és que, al desenvolupar l'aplicació en diferents nivells, sempre que aparegui la necessitat d'algun canvi, només cal modificar el nivell que ho requereixi.

- **Nivell o capa de presentació:** També anomenada capa d'usuari, és la que conté la interfície que l'usuari visualitza. Presenta el sistema a l'usuari, li comunica la informació i captura les dades introduïdes per l'usuari. Aquesta capa es comunica únicament amb la capa de negoci. Ha de tenir la característica de ser amigable, es a dir comprensible i fàcil d'utilitzar.
- **Nivell o capa de negoci:** Rep les peticions d'usuari, es processen i s'envien respostes. En aquesta capa, es troba tota la lògica del programa, així com les estructures de dades y els objectes encarregats de manipular-les. Es comunica amb la capa de presentació per rebre sol·licituds i presentar resultats, i amb la capa de dades per emmagatzemar o sol·licitar dades. Bàsicament és el cor de l'aplicació.
- **Capa de dades:** És la capa on s'emmagatzemen les dades i l'encarregada d'accedir a elles. Està formada per un o més gestors de bases de dades que realitzen l'emmagatzemament, reben sol·licituds i extreuen la informació.

Aquestes tres capes, són la representació lògica d'un sistema complet, però aquest sistema pot tenir formes diverses: un sol exe, components distribuïts en diferents servidors o un lloc web. Sense importar l'aplicació específica, és útil descriure a tots els sistemes en termes d'aquestes tres capes.

I és per aquest motiu, que a l'hora de fer el disseny de l'arquitectura, l'estructura de tres capes o nivells ha sigut la referència. Tot i que actualment les dades i l'accés es realitza de manera local, una de les possibles línies futures, seria dotar a l'aplicació d'una capa de dades que, per exemple, permetés l'accés via web.

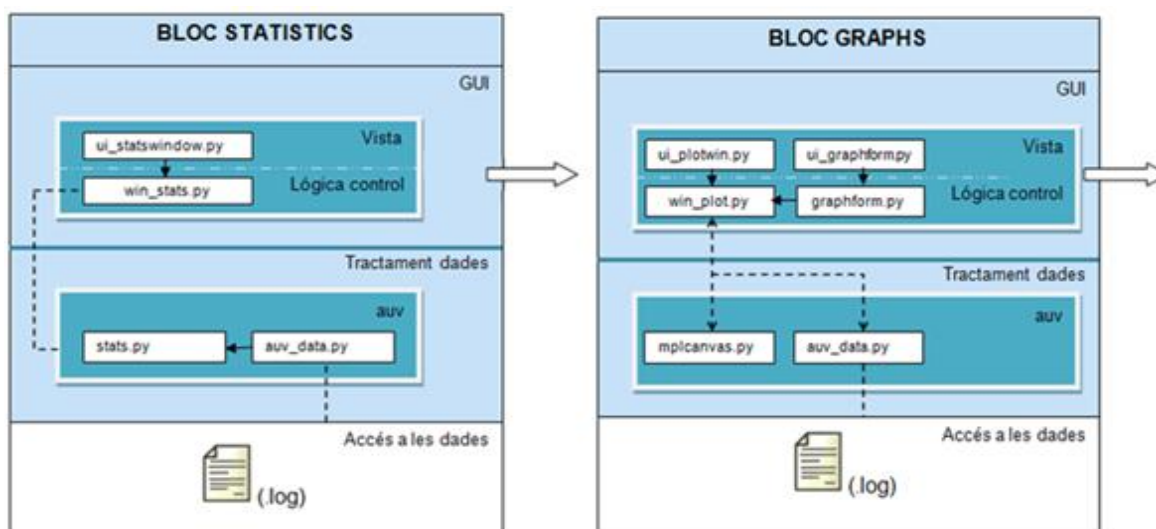
Per altre banda, cal parlar de la modularitat del sistema. Un dels errors comès a l'inici del projecte, ha sigut no visualitzar la magnitud de l'aplicació que s'estava desenvolupant, fins a arribar a tenir una quantitat de paquets i classes considerable. El desenvolupament en base a mòduls, on es separa cada funcionalitat de l'aplicació, ha donat solució a aquesta problemàtica.

El disseny modular ha suposat grans avantatges en les etapes posteriors de disseny i implementació, com per exemple:

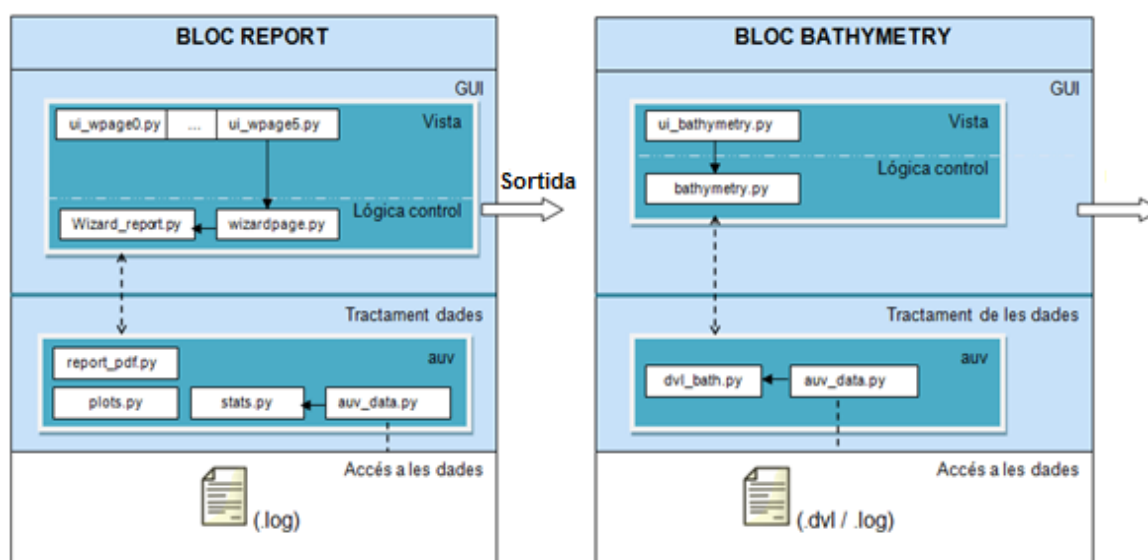
- Facilitat per resoldre els requeriments, i poder testear el seu funcionament de manera independent.
- Claredat alhora de detectar errors dintre de l'aplicació.

Per realitzar l'arquitectura, a part d'especificar el principi que guia el seu disseny, és necessari conèixer l'organització fonamental del sistema, és a dir, cal especificar els components que la formaran i la manera en què es relacionen entre ells.

Dit això, a continuació es mostra de manera esquemàtica l'arquitectura dels diferents blocs que formen el sistema.



**Fig. 3.3** Disseny arquitectura bloc estadístic i bloc gràfic



**Fig. 3.4** Detall de l'arquitectura dels blocs report i batimetria

El sistema final, està format per quatre blocs o subsistemes que actuen i generen resultats de manera independent (Fig. 3.3 i 3.4). Cada bloc, accepta com a entrada de dades un o diversos fitxers d'estat del vehicle (.log), i en el cas del bloc de batimetria, és necessari carregar també les dades provinents del dvl/adcp emmagatzemades en un fitxer d'extensió .dvl. El mòdul `auv_data`, s'encarrega de crear l'estructura de dades per al emmagatzemament d'aquestes i de proporcionar l'accés a la resta de mòduls. Com és pot observa als esquemes de l'arquitectura (Fig. 3.3 i 3.4) aquest mòdul es reutilitza en tots els blocs.

`plots.py`, `report_pdf.py`, `dvl_bath.py`, `stats.py` i `mplcanvas.py`, són els mòduls encarregats d'implementar les diferents funcionalitats descrites anteriorment. Per tant, tota la lògica de l'aplicació està continguda en aquests. Cadascun dels mòduls genera una sortida que pot ser gràfica, report en format pdf, un conjunt de punts representant coordenades (x,y,z), o resultats numèrics estadístics.

Els resultats es mostren a l'usuari de manera totalment transparent, mitjançant la capa de presentació que conté les classes encarregades del disseny visual. La lògica de control, fa la funció de comunicar les peticions de l'usuari a la capa de negoci, i els resultats generats per aquesta a l'usuari.

En resum, el disseny de l'arquitectura del software desenvolupat, s'ha basat en l'estructura de capes, que dota al sistema de la capacitat d'escalabilitat, i fa que una mateixa capa de negoci pugui interactuar amb diferents bases de dades futures i tenir diferents presentacions.

Alhora, el desenvolupament de la capa de negoci, es a dir la lògica de l'aplicació, s'ha encarat de manera modular, on cada mòdul té una responsabilitat atenent a les diferents funcionalitats del sistema, intentant que cadascun d'ells tingués una alta cohesió i un baix acoblament.

### 3.4 Disseny del software

En aquest apartat, s'exposen els criteris que s'han pres alhora de fer el disseny de baix nivell del software, així com els diferents diagrames UML, on queda plasmat amb més detall la relació entre els diferents mòduls i classes que formen el software. Partint de que l'arquitectura del sistema ha guiat tot el procés de disseny, s'ha cregut convenient dividir-ho en dos subapartats: el primer implementa la lògica del sistema, i el següent tracta els aspectes de la interfície gràfica.

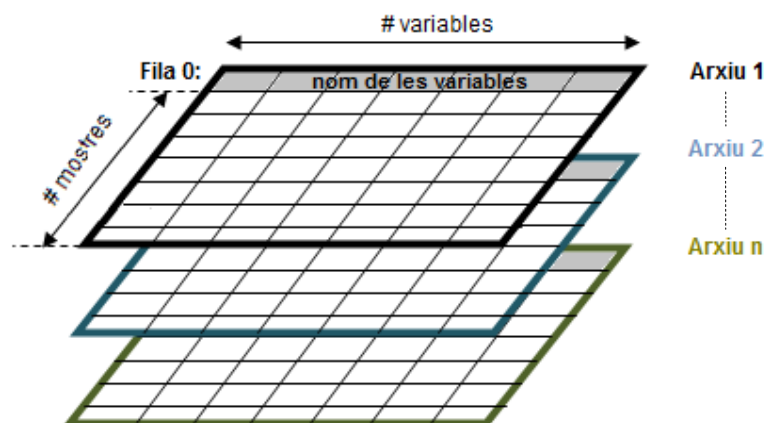
#### 3.4.1 Mòduls per al tractament de les dades

Com a sistema modular, pot ser entès com la unió de diferents parts, que interactuen entre si i que treballen per assolir un objectiu comú, realitzant cadascuna de les parts una tasca necessària per a la consecució de dit objectiu. Idealment, un mòdul ha de complir la funció de caixa negra, és a dir ser independent de la resta de mòduls i comunicar-se amb ells a través d'entrades i sortides ben definides.

##### 3.4.1.1 Mòdul *auv\_data.py*

Conté la classe principal per al tractament inicial de les dades. Defineix l'estructura de dades i dicta l'organització, els mètodes d'accés i les alternatives de processament.

L'estructura dissenyada possibilita la lectura de fitxers d'estat .log i .dvl de manera individual o per paquets i facilita la bolcada de les dades en una BBDD. Està definida com un espai n-dimensional i es caracteritza per ser de tipus diccionari o també anomenada matriu associativa. Dit d'una altra manera, és un conjunt de parelles clau-valor, on la clau ha de ser un tipus de dada immutable i el valor pot ser de qualsevol tipus.



**Fig. 3.5** Estructura de dades de tipus diccionari

La funció `load_data` és l'encarregada de llegir i emmagatzemar el contingut de cada fitxer en una matriu (valor) que està indexada mitjançant el nom del fitxer (clau) (Fig. 3.5).

Els mètodes d'accés a les dades implementats són:

- **get\_variable:** permet buscar una variable en la matriu de dades d'un únic fitxer. Retorna un vector de floats.
- **get\_var\_total:** A diferència del mètode anterior, aquest retorna un vector de floats amb els valors d'una variable, però de tots els fitxers carregats.
- **get\_datetime:** L'estructura de dades conté valors numèrics i valors que representen dates o instants de temps i requereixen un tractament especial. Python compta amb un mòdul especialitzat en el processat de dates: `datetime`. Per tant `get_datetime` s'encarrega de buscar en la matriu, dades de temps i dates, i construir un vector d'objectes de tipus `datetime` que a través dels seus mètodes i atributs permet manipular i formatjar aquest tipus de dades.
- **get\_UTMcoord:** Les coordenades de Latitud i Longitud emmagatzemades al fitxer d'estat del vehicle venen donades en graus decimals. Per tal de adequar el tipus de dada per a posteriors càlculs, com per exemple la distància recorreguda pel vehicle, és necessària la conversió a coordenades UTM (unitats en metres). Així doncs aquest mètode retorna els vectors de Latitud i Longitud transformats a coordenades UTM.

#### 3.4.1.2 Mòdul *Stats.py*

El mòdul `stats.py` implementa els algorismes per al càlcul dels diferents paràmetres estadístics, com màxims, mínims i mitja de diferents variables, distància total i de submersió recorreguda pel vehicle, temps de missió, temps submergit, eficiència etc.



**Fig. 3.6** Entrada i sortida del mòdul `stats.py`

Rep com a entrada un objecte de tipus `auvData`, que conté l'estructura amb les dades a analitzar i proporciona els mètodes d'accés a les variables d'interès. Genera els resultats, i els emmagatzema en una matriu n-dimensional.

### 3.4.1.3 Mòdul *Mpl\_canvas.py*:

Defineix la classe Canvas per generar una gràfica amb el suport de les llibreries matplotlib i numpy.

Matplotlib, s'utilitza habitualment per a la representació de grans conjunts de dades. Les llistes o *arrays* nadius de Python són extremadament flexibles y pràctics, però quan es tracta d'un gran número d'elements, mostren els seus límits. Per tals mesures, matplotlib necessita un major suport. Numpy és el mòdul estàndard de python per elaboracions numèriques, que proporciona suport a les operacions d'alt rendiment, amb tipus de dades com *arrays* i matrius, juntament amb moltes funcions matemàtiques. Un dels aspectes fonamentals de Numpy és que proporciona un potent objecte de tipus array N-dimensional, *ndarray*, per representar un conjunt d'elements (tots del mateix tipus).

Abans de continuar amb la definició de la classe Canvas, cal introduir breument quins objectes de matplotlib componen una figura. Des del nivell superior al inferior els objectes estan enriats de la següent manera:

- **FigureCanvas**: contenidor per la instància de la figura.
- **Figure**: contenidor per una o més instàncies d'eixos.
- **Axes**: Àrees rectangulars per contenir els elements bàsics, tals com línies, text y així successivament.

Per tant, per poder representar les gràfiques amb Matplotlib, s'ha definit la classe Canvas. Aquesta hereta de la classe FigureCanvas, que a banda de ser una classe de Matplotlib, és també un QWidget – la classe base de tots els objectes d'interfície d'usuari. Això significa que es pot tractar a FigureCanvas com un objecte QWidget i utilitzar-lo com un element més en la finestra principal de l'aplicació. D'aquesta manera la classe té com atributs un objecte de tipus Figure, un axes, on es podrà dibuixar, i dos vectors x, y, els quals contendran els valors a representar. El mètode `_init_` és l'encarregat d'instanciar i inicialitzar els objectes. També s'instancia FigureCanvas, que s'encarrega de prendre l'objecte Figure de Matplotlib i presentar-ho com un objecte QWidget, per poder-lo fer servir en una finestra de la GUI.

Els mètodes `time_plot`, `samples_plot`, `compar_plot` i `text_properties` s'encarreguen de representar els punts dels vectors x, y, i modificar l'aparença de la figura. Els canvis en la figura és realitzen a través de comandes del mòdul Matplotlib.pyplot. El motor subyacent de pyplot manté un registre de l'estat de la figura activa i de la informació de l'àrea de dibuix, i a través de les funcions de *plotting* canvia aquesta informació. Per fer-ho més clar, els mètodes implementats per dibuixar a la figura, no utilitzen referències a l'objecte durant el procés, sinó que a través de les comandes pyplot modifiquen la informació i els canvis apareixen en la figura. És pot entendre com una màquina d'estat, és a dir un sistema amb un estatus global, on cada operació realitzada sobre el sistema canvia el seu estatus.



#### 3.4.1.4 Mòdul *Plots.py*

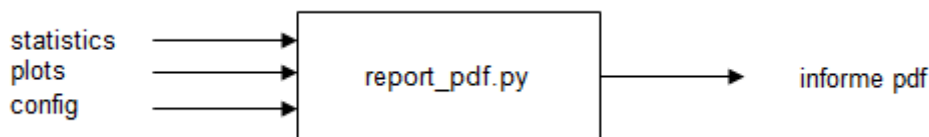
Defineix les funcions per crear un conjunt de gràfiques amb l'ús de les llibreries Matplotlib i numpy. A diferència del mòdul *mplCanvas*, aquest no s'utilitza per a la representació dinàmica en una finestra de l'aplicació, sinó que el conjunt de gràfiques resultant d'aquest mòdul s'inclouran en el report pdf.

Com qualsevol programa, Matplotlib té una configuració per defecte i per tant s'ha de configurar segons les pròpies preferències. Hi ha diferents maneres per controlar la personalització, des de modificar l'arxiu de configuració global o crear-ne un d'usuari, a utilitzar codi Python en el programa actual. En aquest cas s'ha optat per l'opció de codi Python perquè els canvis només es volen a nivell local, i no que afectin a tots els usuaris de la màquina. Per tant, cada funció implementada, s'encarrega de definir les seves pròpies característiques de l'espai de representació segons la gràfica o gràfiques a dibuixar. Els paràmetres personalitzats són el tipus de línia a utilitzar, la resolució dels eixos la informació a les gràfiques com llegendes, títols, nom dels eixos, tipus de font etc.

Cada funció rep com entrada un objecte de tipus *auvData*, i la ruta d'un directori temporal on s'emmagatzemaran els arxius de sortida d'extensió *.png* generats per aquest mòdul.

#### 3.4.1.5 Mòdul *Report\_pdf.py*

Defineix les funcions de disseny i maquetació per crear un informe en format pdf. Aquest inclou una fitxa tècnica del projecte, estadístiques de campanya, i un conjunt de gràfiques d'interès. La figura 3.7 mostra les entrades i sortida d'aquest mòdul.



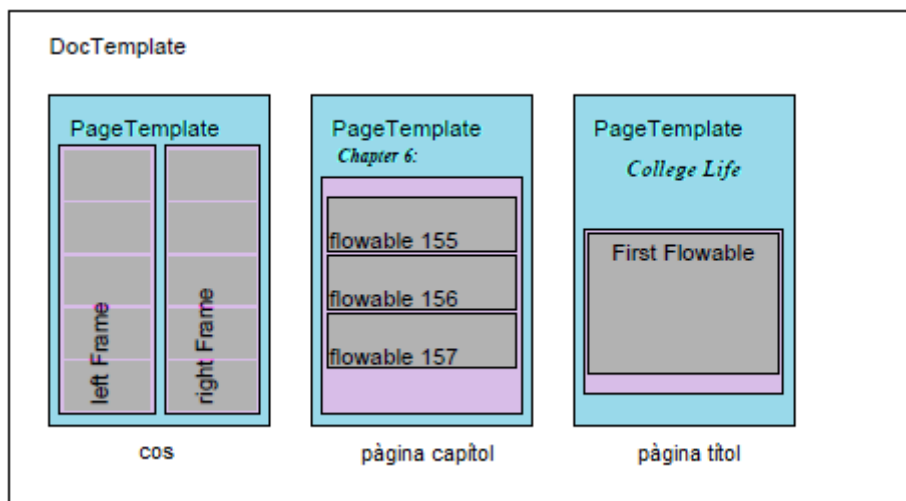
**Fig. 3.7** Esquema d'entrades i sortides del mòdul *report\_pdf*

Per implementar les funcions que generen el report és fa servir la biblioteca ReportLab especialment el mòdul Platypus (Page Layout and Typography Using Scripts). És una llibreria d'alt nivell de disseny de pàgines la qual mitjançant programació pot crear documents complexes.

Amb aquesta eina es busca separar les decisions de disseny del contingut del document, tant com sigui possible. És a dir, que permet canviar l'aparença del document sense haver de modificar res del contingut o del text.

Platypus està pensat per fer un disseny per capes, de dalt cap a baix. Les diferents capes en que està estructurat el disseny són:

- **DocTemplates:** És el contenidor últim del document.
- **PageTemplates:** Especificacions pel disseny de pàgines de varies classes.
- **Frames:** Especificacions de regions en les pàgines que poden contenir text o gràfics.
- **Flowables:** Elements de text o gràfics que estan incrustats en el document ( com imatges, paràgrafs o taules.)
- **Pdfgen.Canvas:** És el nivell més baix que en última instància rep ordres de pintar en el document des de les altres capes.



**Fig. 3.8** Il·lustració de la estructura de DocTemplate

L'estil del document es determina doncs mitjançant DocumentTemplates. Cada DocumentTemplate conté un o més PageTemplate que defineix l'aspecte de les pàgines del document. Alhora cada PageTemplate compta amb procediments per l'elaboració de la part fixa de la pàgina com la capçalera, peu de pàgina, logotip, etc. i procediments per la definició dels marcs que encapsulen la part fluida de la pàgina (paràgrafs, taules, imatges etc.).

Les pàgines es construeixen fent servir plantilles (templates) de pàgina, i els paràgrafs de text mitjançant estils de paràgraf amb la intenció de poder reformatjar documents amb poques línies en un únic fitxer, el qual conté els estils i les especificacions de disseny de la pàgina. Un símil podria ser una pàgina HTML i el disseny que se li dóna mitjançant una fulla CSS.

Inicialment en el mòdul pdf report, s'especifiquen els estils de paràgrafs. Aquests s'organitzen en un objecte de tipus diccionari. Amb les funcions `first_page`, `later_pages`, es defineixen les característiques fixes de la primera pàgina del document i de la resta de pàgines. Aquestes dues funcions utilitzen operacions del nivell més baix `pdfgen.canvas` per pintar a la pàgina la capçalera, peu de pàgina i els logos.

Un cop definides les plantilles, en la funció `generate_report`, es crea un document des de la classe `DocTemplate`, i se li passa una llista `Flowables` mitjançant el mètode `build`. Internament, el mètode `build` del document, processa la llista de `Flowables` de manera raonable fent servir diferents events.

La resta de funcions implementades, s'utilitzen per construir la seqüència d'elements bàsics `flowables`, *Platypus story*. Aquestes funcions són: `Project_technical_details`, `tecnic_stats`, `scientific_stats`, `navigation_graphs`, `general_graphs`, `scientific_graphs`, `user_graphs`, `files_include`.

Cadascuna d'elles implementa una secció del document, fent servir principalment la classe `Paragraph`, que dona format al text i estableix l'estil de la font i els canvis de color mitjançant el llenguatge de marcat XML. Per inserir les gràfiques es fa servir la classe `Image` de *Platypus*, que crea un `flowable` a partir d'una imatge definida en un arxiu de tipus `.png`.

La figura següent, mostra com mitjançant les funcions definides es va creant el *story* a partir de la instància de la classe `Paragraph`.

```
def scientific_stats(stats, story):
    stats_results = stats.stats_results
    story.append(Spacer(1,1*cm))
    story.append(Paragraph("<b>INITIAL DATE: </b>%s <b>END DATE:</b> %s</br>"
        %(stats_results[0][0],stats_results[0][1]), normal_indented1_style))
    story.append(Paragraph("<b># LANZAMIENTOS:</b> %d</br>"%(stats_results[1][0]),
        normal_indented1_style))
```

**Fig. 3.9** Fragment de codi de la funció `scientific_stats`

#### 3.4.1.6 *Bathymetry.py*:

Implementa l'algoritme descrit a la secció 2.2.3. Rep com a entrada dos objectes `auvData` amb les dades generades per l'auv (`.log`) i les de l'adcp (`.dvl`) i genera un arxiu de sortida d'extensió `.txt` amb les coordenades corregides dels punts d'impacte dels diferents feixos amb el fons marí. Les funcions principals d'aquest mòdul són:

- **get\_beamrange**: Transforma les dades de profunditat del feix en la longitud obliqua del feix, donat l'angle de inclinació del transductor. Les dades de profunditat dels feixos proporcionades per l'adcp, són les distàncies verticals entre la superfície i la ubicació on el feix impacta amb

el terra. Rep com a entrada quatre vectors amb les distàncies verticals de cada feix, i retorna aquest vectors transformats.

- **get\_offsets:** Implementa la matriu de rotació i calcula els offsets o compensacions que cal fer a cada punt dels diferents feixos tenint en compte el angles de roll, pitch i heading. Rep com a entrada els valors de longituds obliqües dels feixos i els de roll, pitch i heading amb els angles en graus decimals, i retorna una matriu amb el valor dels offsets (en m.) de cada feix. La matriu té una dimensió  $n \times m$ , on  $n$  és 3 (offset de la coordenada  $x, y, z$ ) i  $m$  és el número de feixos, en aquest cas, quatre.
- **generate\_bathymetry:** Processa les dades provinents de l'auv i de l'adcp i genera els resultats batimètrics de cada mostra (coordenades  $x, y, z$  corregides dels punts d'impacte dels quatre feixos amb el fons). Retorna tres vectors  $x, y, z$ , on  $x$  i  $y$  contenen tots els valors de latitud i longitud expressats en coordenades UTM i  $z$  la profunditat en m.
- **export\_to\_file:** Genera un fitxer d'extensió .txt amb els resultats batimètrics obtinguts. Cada línia conté la informació de latitud, longitud i profunditat dels punts separats per un espai.
- **plot:** Aquesta funció obre el fitxer de sortida amb les coordenades batimètriques i les representa en una gràfica.

### 3.4.2 Interfície gràfica (GUI)

Com ja s'ha anat repetint al llarg d'aquest projecte, l'usuari del software probablement no serà un expert en informàtica, per tant la manera d'interactuar amb el software ha de ser mitjançant una interfície gràfica que l'ofereixi la possibilitat de realitzar les accions desitjades de forma senzilla, ràpida i intuïtiva.

El software està format per un conjunt de formularis independents que permeten a l'usuari establir un diàleg amb l'aplicació. Cada bloc mostrat a la part d'arquitectura consta d'un o més formularis per mostrar els resultats dels mòduls funcionals i per atendre les peticions de l'usuari. En el cas del report, en lloc de dissenyar una finestra, s'ha definit un assistent, que és un tipus especial de diàleg d'entrada format per una seqüència de pàgines. El propòsit d'aquest, es guiar a l'usuari a través de les diferents passes necessàries per generar el report. Una finestra principal amb un menú, és l'encarregada d'englobar totes les funcionalitats sota un mateix entorn.

Python compta amb gran varietat de llibreries gràfiques per al disseny de les interfícies. En aquest cas, es fa servir PyQt4, exposada anteriorment, i Qt Designer per crear les interfícies gràfiques, un editor de disseny que ha facilitat en gran part la tasca.

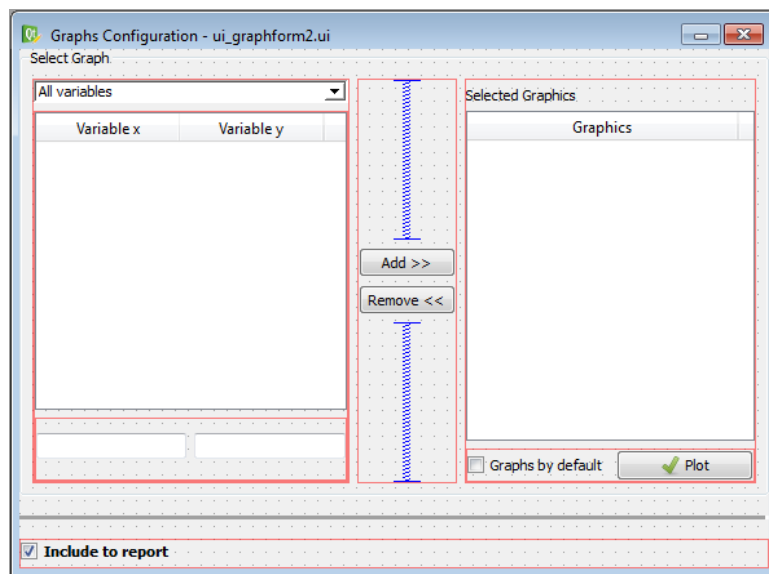
Els formularis editats amb Qt Designer són generals, i no específics per a cada llenguatge. Per aquest motiu, ha sigut necessari compilar els arxius de sortida

del Qt Designer d'extensió .ui a través de la següent comanda des de consola:

```
pyuic4 -o nom_arxiu.py -x nom_arxiu.ui
```

que el què fa és convertir un arxiu xml de definició del formulari, en formularis usables des de Python.

Cada finestra es sustenta en una classe independent, com es pot veure en el diagrama UML (apartat 3.5).i té com atributs diferents objectes, com per exemple, QPushButton, QTableWidgetItem o QVBoxLayout, components del mòdul PyQt.QtGui.



**Fig. 3.10** Finestra de l'aplicació creada amb QtDesigner

Tot i no ser la part més important de l'aplicació, el repte de dissenyar la interfície gràfica, ha sigut una part amb força rellevància dintre d'aquest projecte, no la part purament visual, sinó la part de lògica de control, o dit d'una altre manera, la interacció sistema-usuari. Qt Designer, no s'encarrega de la lògica de control, per tant ha fet necessària la seva implementació i el previ disseny de diagrames de flux per tal de controlar tots els processos. La lògica de control, permet independitzar encara més, la capa visual de la lògica de negoci.

Per a cada funcionalitat, s'ha desenvolupat una classe que conté el conjunt d'operacions lògiques i estructures de control per determinar l'ordre d'execució de les instruccions del programa.

Per no estendre excessivament el document, no és creu convenient detallar les funcions de cada classe implementada però si mencionar la gestió d'events, ja què és la base de la lògica de control.

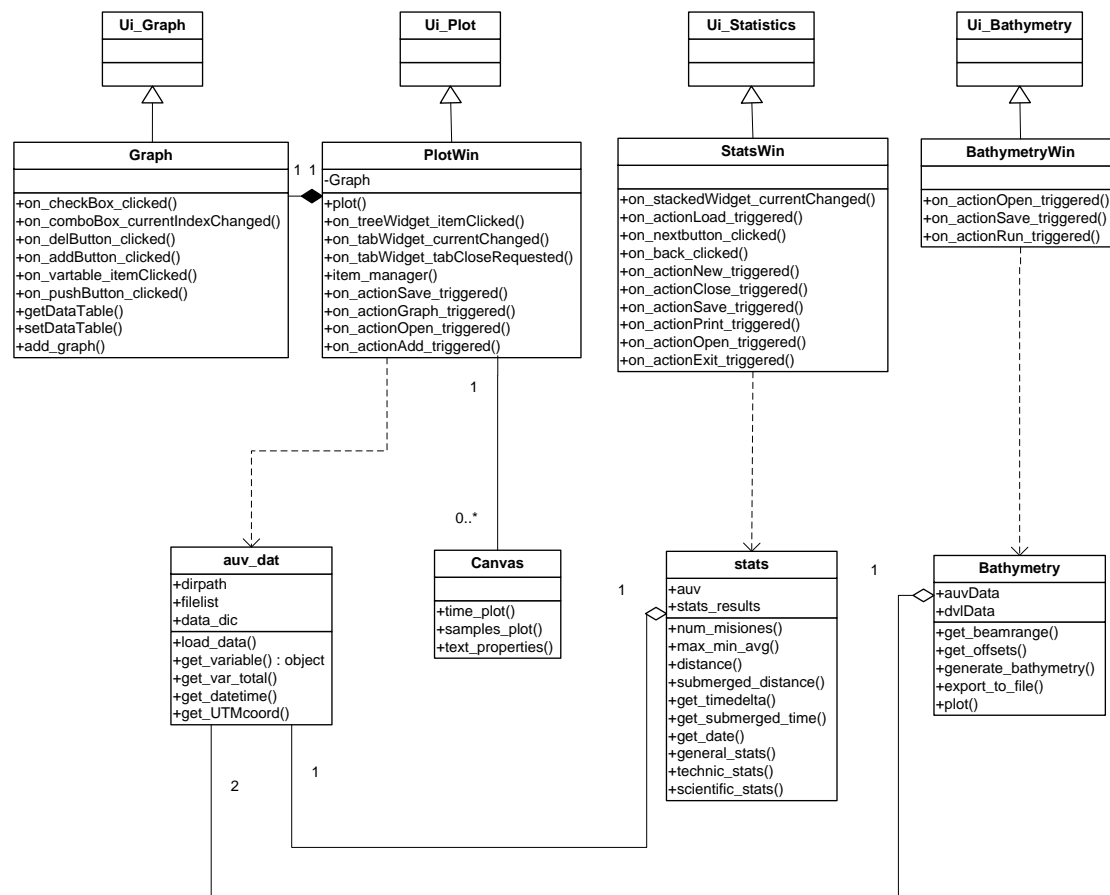
Qt va introduir el sistema Signals & Slots que resulta fàcil d'utilitzar per controlar els events de l'aplicació i és un dels conceptes fonamentals. Cada component de la finestra (objecte), emet una senyal quan succeeix un event. Per exemple un botó emet una senyal clicked quan aquest es clicat. Aquesta senyal es connecta a un o més slots, que són les accions a prendre quan la senyal s'emeti. Existeix la possibilitat d'encadenar senyals, o d'aplicar més d'un slot a una senyal etc. Senyals i slots s'utilitzaven inicialment per a la gestió d'events, però també es pot utilitzar per deixar que els objectes interactuïn. (tenint aquest fet ocult per ells). Aquest objectes no saben res sobre qui emet la senyal o on estan els slots. Tot està gestionat per Qt, el que vol dir que estan dèbilment acoblats, resultant d'aquesta manera components reutilitzables. Aquest sistema ha dotat l'aplicació de robustesa i ha permès un control del cicle d'execució de l'aplicació.

També mencionar que s'ha optat per la gestió d'excepcions en front a la tàctica de comprovar precondicions a priori. Les excepcions son errors detectats per Python durant l'execució del programa. Quan l'interpret es troba en una situació excepcional, com per exemple dividir un número entre zero o intentar accedir a un arxiu no existent, aquest genera o llança una excepció, informant a l'usuari que existeix un problema. Si l'excepció no és captura el flux d'execució s'interromp i l'aplicació es penja o es tanca.

Per tant, per a la gestió d'excepcions s'ha utilitzat l'estructura de control try/except que ofereix Python. L'estructura try/except permet capturar les excepcions ocorregudes durant l'execució del programa i realitzar una determinada acció per a cada tipus d'error. El bloc try tanca el fragment de codi en què podria produir-se una excepció. El bloc except permet indicar el tractament que es portarà a terme si es produeix dita excepció. Molts cops aquest tractament ha consistit simplement en imprimir un missatge informatiu indicant a l'usuari que ha realitzat una acció incorrecta i tornar al punt d'execució previ a l'error, d'altres s'ha utilitzat per registrar el tipus d'error i quan ha sigut possible, s'han establert estratègies de resolució del problema. Python permet utilitzar diversos excepts per un sol bloc try de manera que podem donar un tractament diferent a l'error segons el tipus d'excepció que ha saltat. També conta amb una clàusula finally que sempre s'executa abans de sortir de la sentència try, tant si s'ha llançat una excepció com si no. Per exemple ha sigut interessant a l'hora de tractar amb els fitxers, d'aquesta manera s'assegura que encara que hi hagi un error processant les línies del fitxer, aquest sempre es tancarà i no quedarà corrupte.

### 3.5 Diagrama de classes UML

El següents diagrames UML mostren de manera visual, les relacions estructurals i d'herència entre les classes del sistema, així com els atributs i operacions de les classes.



**Fig. 3.11** Diagrama de classes dels blocs Bathymetry, Plots i Statistics

En la figura 3.11 és pot veure el diagrama de classes resultant de la implementació dels blocs de batimetria, gràfiques i estadístiques. Conté totes les classes d'elements gràfics que s'utilitzen en la implementació i disseny de la interfície gràfica del software. Aquestes són *Ui\_Graphs*, *Ui\_Plots*, *Ui\_Statistics* i *Ui\_Bathymetry*. També les classes que suporten la lògica de control i implementen el flux d'execució de l'aplicació. Aquestes hereten directament de les classes gràfiques i comuniquen la capa visual amb la lògica de negoci o classes funcionals. I per últim les classes funcionals que són les que suporten tota la lògica de negoci.

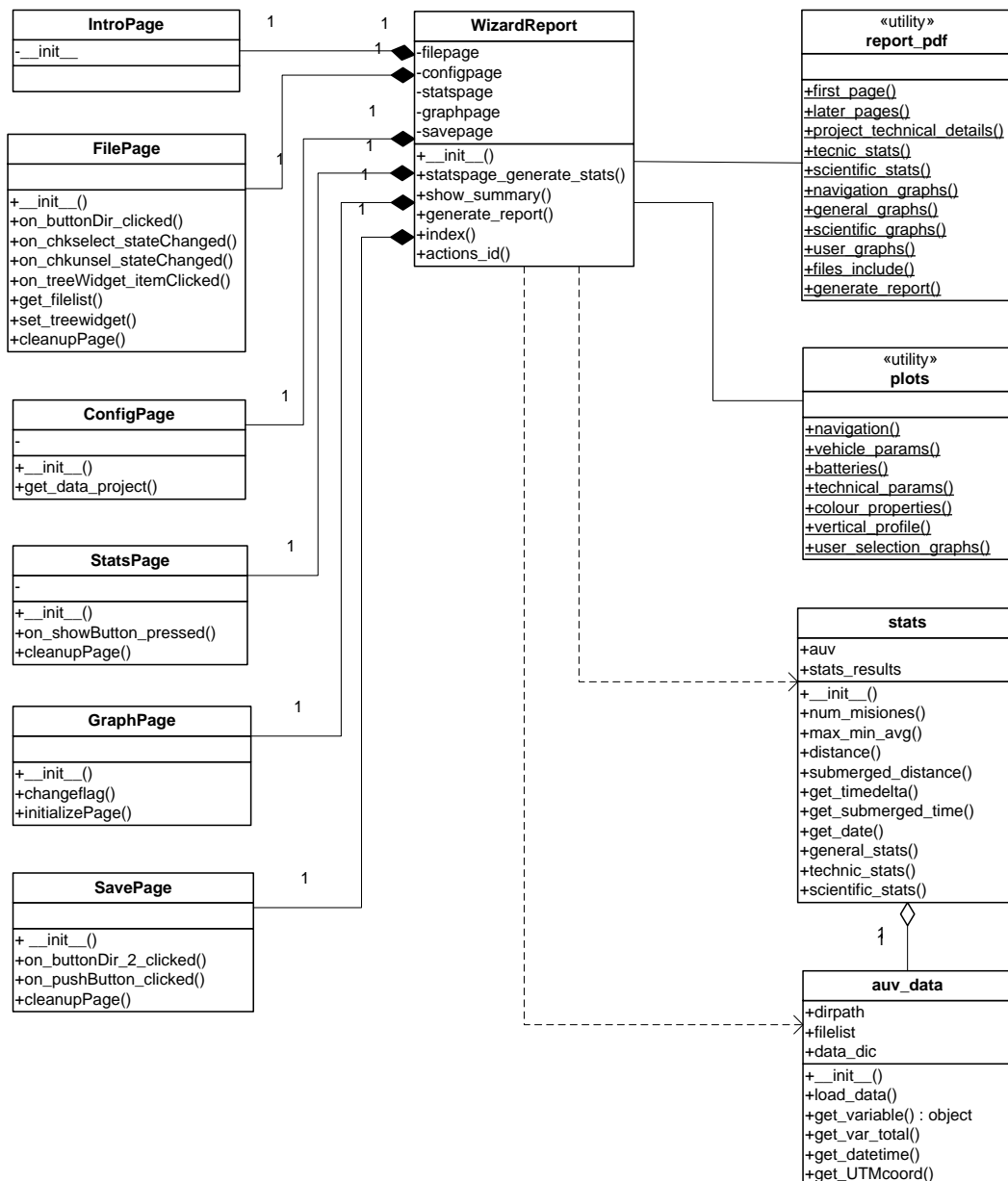


Fig. 3.12 Diagrama UML bloc Report

La interfície gràfica del mòdul report s'ha implementat, com ja s'ha comentat, mitjançant un assistent fent servir la classe **QWizard** que hereta directament de **QDialog** i és la classe base de les finestres de diàleg. Per tenir un major control, cada pàgina es sustenta en una classe independent que hereta de **QWizardPage** (una subclasse de **QWidget**).



## CAPÍTOL 4. RESULTATS

Al llarg del desenvolupament d'aquest treball, s'ha analitzat la problemàtica que existeix amb el tractament de dades oceanogràfiques, com s'adquireixen a través d'un vehicle autònom submarí i com s'emmagatzemen. A més s'ha estudiat l'alternativa de com realitzar una batimetria del terreny mitjançant un AUV, a partir d'un correntòmetre d'efecte Doppler.

La revisió d'aquestes premisses ha permès establir les bases per poder realitzar el modelat i la implementació d'un producte de software com és *auvSoft*, que constitueix una eina útil en el processat i gestió de dades oceanogràfiques, amb un component visual capaç de generar gràfiques, estadístiques i una batimetria senzilla del terreny. Alhora que ofereix la possibilitat de crear informes de manera automatitzada i d'aquesta manera ajuda a planificar i optimitzar el desplegament de futures campanyes de recollida de dades.

La manera en que ha estat dirigit el procés de desenvolupament ha fet possible obtenir un producte amb les següents característiques generals:

- Diverses funcionalitats, agrupades en un mateix entorn.
- Components per a la visualització i anàlisi de les dades: accés a la informació mitjançant cerques en el domini espacial paramètric amb sortides gràfiques i numèriques comprensibles.
- Components per a la publicació de resultats.
- Una interfície senzilla, fàcil d'utilitzar, atractiva i funcional.
- Transparència: L'usuari no ha d'aprendre detalls de baix nivell de les dades tractades.
- L'arquitectura i la manera en què s'ha modelat el sistema fan que aquest sigui escalable, és a dir posseeix la capacitat d'expansió en quant a funcionalitats. Per això aquest treball es pot considerar com un punt de partida per al desenvolupament de nous projectes més ambiciosos.
- Tot i que el software s'ha desenvolupat sota Windows, permet la portabilitat a altres plataformes gràcies al llenguatge utilitzat i a l'elecció de llibreries multiplataforma

### **Instal·lació del software:**

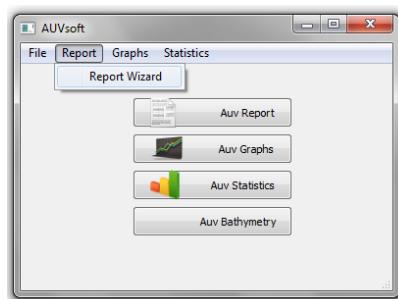
Per facilitar la distribució convindria generar un sol .exe amb totes les dependències incloses, tanmateix, l'aplicació resultant d'aquest projecte no contempla aquesta possibilitat i fa necessària la instal·lació dels següents paquets i llibreries per a l'ús del software:

- Python 2.7
- PyQt4
- Matplotlib /Numpy
- Reportlab

Un cop instal·lat, només cal executar l'arxiu mainwindow.py per fer córrer l'aplicació.

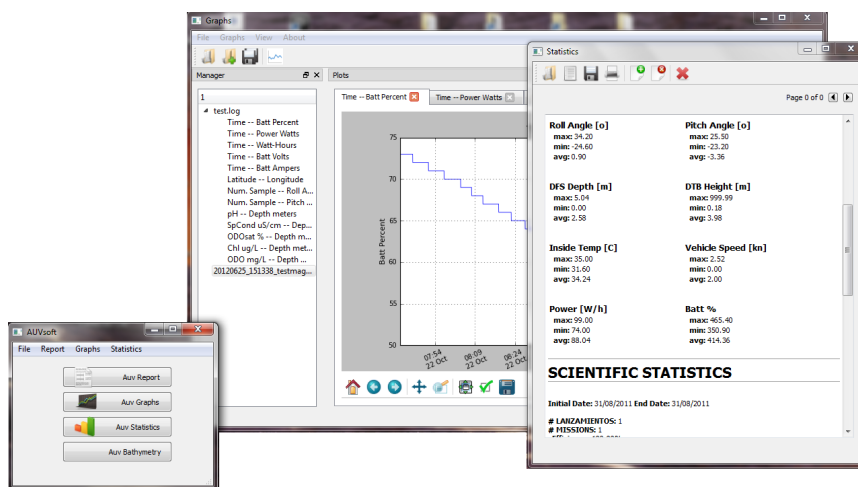
## 4.1 Aspectes generals de l'aplicació

Al executar el programa, s'inicia un menú principal, que ofereix a l'usuari la possibilitat d'escollir entre les diferents funcionalitats dintre d'un mateix entorn.



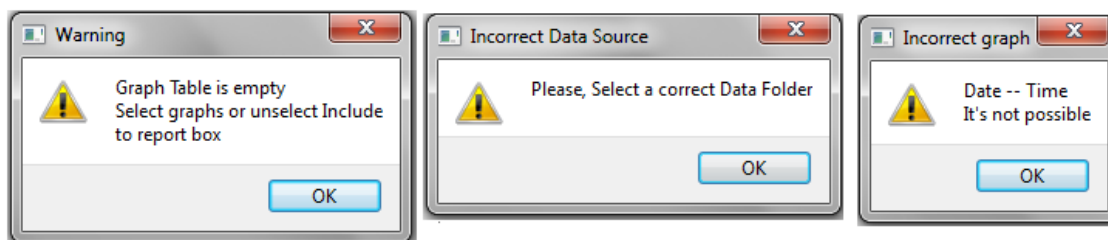
**Fig. 4.1** Finestra inicial de l'aplicació

El menú superior de la barra d'eines, obre una finestra independent i permet a l'usuari realitzar diverses funcions paral·lelament. Aquesta característica és d'especial interès, ja que ofereix la possibilitat de comparar els resultats estadístics amb els visuals, o representar alhora la batimetria del terreny i la gràfica de la trajectòria del vehicle, profunditat, etc. i d'aquesta manera obtenir una visió més compacta de les dades analitzades.



**Fig. 4.2** Simultaneïtat de funcions

Pel que fa a la robustesa de l'aplicació, el flux d'execució del programa està controlat mitjançant el llançament de missatges d'alerta en cas de realitzar alguna operació incorrecta, i la gestió d'errors en temps d'execució.



**Fig. 4.3** Exemple de diferents missatges d'alerta de auvSoft

## 4.2 Funcionalitats:

La forma d'avaluar l'acceptació del software ha consistit en la validació del requeriments establerts en la fase d'anàlisi i disseny. Clar que per això ha sigut necessari considerar els criteris d'avaluació de cadascun d'ells per separat. Això ha donat com a resultat que la eina desenvolupada presenta un bon nivell de robustesa i acceptació en quant a la realització de les funcions següents:

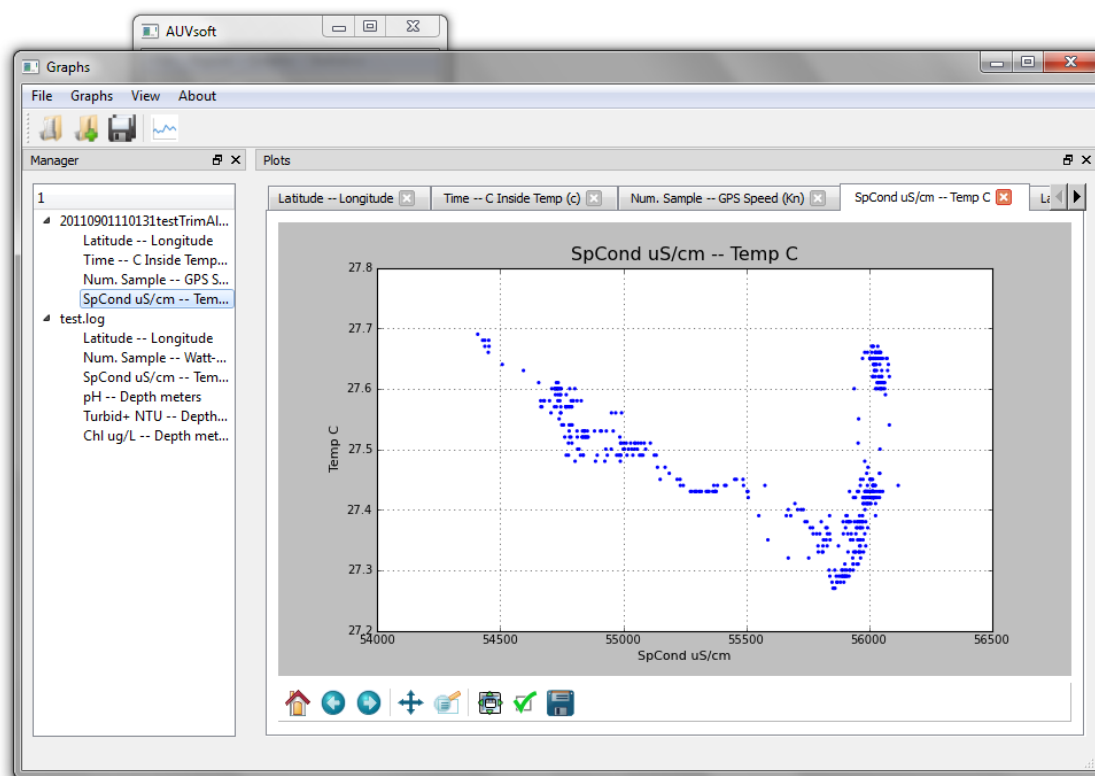
### 4.2.1 Monitoratge de variables

La visualització gràfica de les dades permet avaluar la seva coherència i detectar el correcte funcionament de l'AUV i dels sensors integrats.

A més permet visualitzar de manera ràpida l'evolució dels diferents paràmetres, i localitzar els punts d'especial interès. Per exemple, és pot observar com les propietats de l'aigua evolucionen en el temps realitzant series temporals, comparar dues propietats del aigua o realitzar perfils verticals de qualsevol propietat, és a dir representar una variable en funció de la profunditat. Aquesta gràfica és especialment útil per determinar l'espessor de les masses d'aigua o localitzar la profunditat on és troben els màxims de clorofil·la, etc.

Inicialment l'usuari selecciona un o diversos fitxers .log d'on s'extrauran les dades a representar. Tant bon punt com les dades s'han carregat a l'estructura, es pot posar en funcionament la representació gràfica. En la barra d'eines s'habilita la icona de representació gràfica. Al seleccionar-lo s'obre un petit formulari on l'usuari pot escollir el parell x-y entre les variables disponibles, i formar un llistat de gràfiques a visualitzar. Un cop escollida la llista de gràfiques, aquesta s'envia al formulari principal indicant d'aquesta manera que és vol iniciar el procés de representació.

Amb l'objectiu de poder tenir un control sobre la informació que s'està visualitzant, l'aplicació compta amb un administrador d'arxius i gràfiques, com es pot veure a la figura 4.4 (banda esquerra).



**Fig. 4.4** Finestra principal representació gràfica.

Les diverses gràfiques es poden emmagatzemar amb un format de sortida .png per a l'ús posterior.

#### 4.2.2 Generació d'estadístiques de campanya

La qualitat de les mesures oceanogràfiques recollides pels sensors, està estretament lligada amb l'estat dels dispositius electrònics que poden patir falles de calibratge, transmissió o configuració que provoquen dades inconsistents.

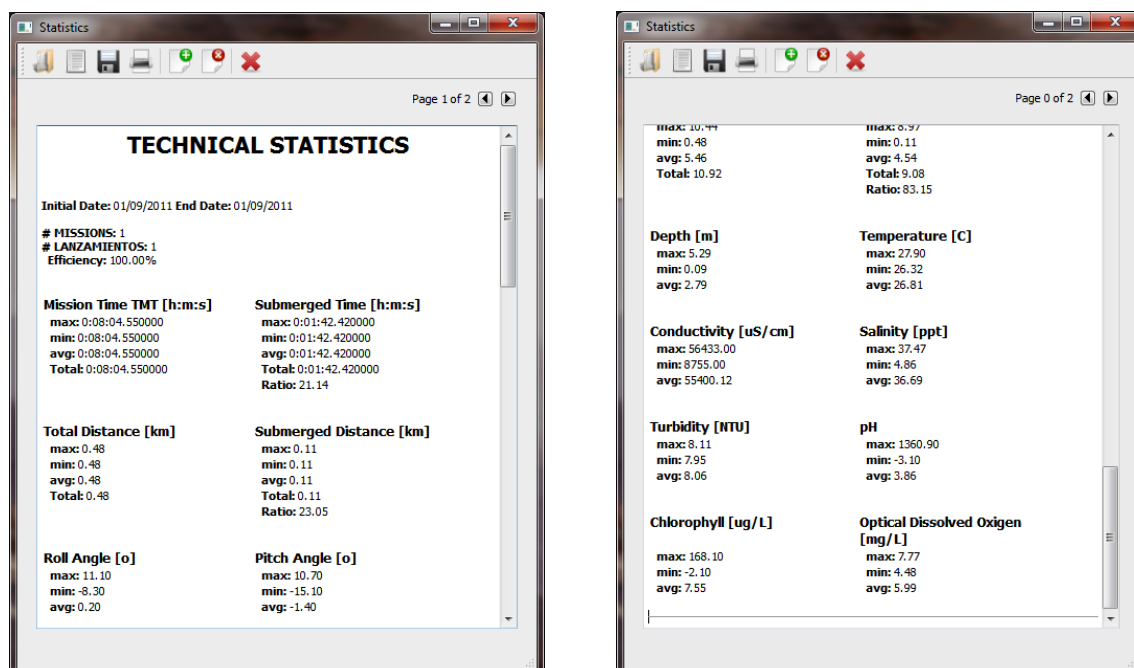
L'estudi estadístic permet realitzar un primer anàlisi i interpretació de les dades d'una mostra representativa, ja sigui per ajudar a prendre decisions o per explicar condicions regulars o irregulars de les variables d'estudi.

Per aquest motiu el software compta amb un mòdul de càlcul d'estadístiques bàsiques de campanya, tant de paràmetres fisicoquímics com de variables d'enginyeria associades a l'estat del vehicle.

En aquest mòdul, es calcula el valor màxim, mínim i la mitja de cada conjunt de dades, com per exemple de la conductivitat, clorofil·la, temperatura de l'aigua, etc. I paràmetres d'enginyeria com nivells de bateries, velocitat del vehicle o temperatura interna entre d'altres. Aquesta estadística permet veure si les

dades son internament consistents, es a dir, amb el seu propi perfil, o amb la comparació dels valors d'un altre dia. També mostra una estadística bàsica de la missió, amb paràmetres com la distància recorreguda en superfície i submergit, el temps de missió etc.

La interfície dissenyada permet a l'usuari generar i visualitzar els resultats estadístics dels fitxers seleccionats, amb la possibilitat tant de generar estadístiques de missions individuals, és a dir, d'un únic arxiu .log com de diverses missions, i fins a una campanya completa.



**Fig. 4.5** Interfície pel càlcul estadístic mostrant els resultats d'una missió.

L'aplicació permet realitzar diferents operacions amb els resultats, com exportar-los a un fitxer de text, imprimir, generar un pdf i les opcions comuns d'un editor de text (copiar, pegar etc.).

Ahora, compta amb una opció per crear noves pàgines i d'aquesta manera permet visualitzar estadístiques de diferents fitxers simultàniament mitjançant l'ús de diverses pàgines.

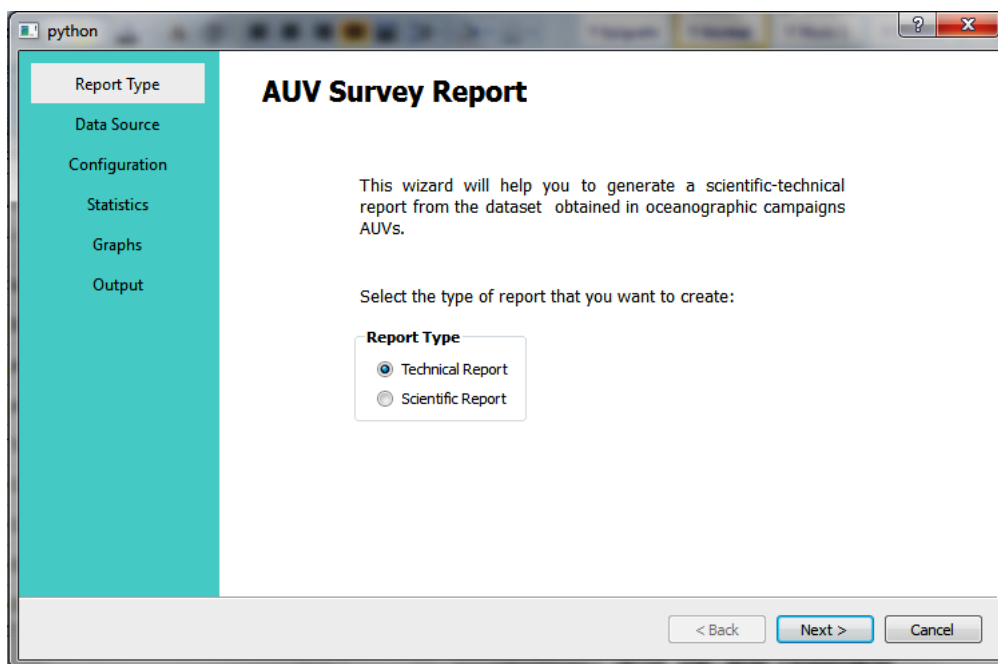
#### 4.2.3 Generació automatitzada de reports científics i tècnics

Després de realitzar una campanya oceanogràfica de recollida de dades, arriba el moment d'analitzar-les i treure'n conclusions.

Donat el volum d'informació que proporciona un vehicle autònom, aquesta tasca pot convertir-se en un treball ardu i lent, i el temps és un recurs molt

valorat, especialment en investigació, com per dedicar-lo a la gestió de la informació. Tot i que no deixa de ser un pas necessari en qualsevol estudi observacional. Amb el desenvolupament d'aquesta eina s'espera millorar el temps en què es realitza aquesta operació.

L'implementació del bloc de report dóna com a resultat un assistent on el flux de treball està automatitzat, es fàcil e intuïtiu d'utilitzar i simplifica el procés de creació de reports en cinc passes. En la figura 4.6 es mostra la interfície inicial d'aquest assistent.



**Fig. 4.6** Assistent per a la creació d'un informe

Es pot escollir entre dos tipus de report, un científic i un altre tècnic segons l'interès de l'estudi a realitzar. El report tècnic es centra en l'anàlisi de paràmetres d'enginyeria del vehicle, nivell de les bateries, velocitat del vehicle, temperatura interna, etc. mentre que el científic analitza variables de tipus fisicoquímic, com pot ser la concentració de clorofil·la, oxigen dissolt, la conductivitat de l'aigua, etc.

L'usuari selecciona la font de dades a partir d'un diàleg on ha d'escollir el directori que conte els arxius d'estudi, i seguidament l'assistent mostra un arbre amb els fitxers d'estat disponibles, permetent seleccionar un, tots o diversos arxius per realitzar l'anàlisi.

Una altra entrada de dades, prové de les metadades que introdueix l'usuari. Aquestes metadades són de tipus descriptiu, i fan referència al context on s'emmarca la campanya de recollida de dades realitzada (Fig. 4.7).

The screenshot shows a software window titled 'python' with a subtitle 'Project Technical details' and the instruction 'Set project parameters'. On the left is a teal sidebar with a list of menu items: 'Report Type', 'Data Source', 'Configuration' (highlighted), 'Statistics', 'Graphs', and 'Output'. The main content area is divided into three sections. The 'Report' section contains a text field for 'Author' with the value 'This is a test'. The 'Project' section contains several text fields: 'Name' (This is a test), 'Acronym' (This is a test), 'Ref' (This is a test), 'Organization' (This is a test), 'Abstract' (a multi-line text area containing five instances of 'This is a test'), and 'Keywords' (This is a test). The 'Survey' section contains two date pickers: 'Initial date' (01/01/2012) and 'End date' (01/01/2012). At the bottom right are three buttons: '< Back' (disabled), 'Next >' (active/highlighted), and 'Cancel' (disabled).

**Fig. 4.7** Configuració de la fitxa tècnica del report.

El disseny i el format del report estan preestablerts. Però l'usuari té un cert grau d'independència a l'hora de crear el report. En el pas quatre, existeix l'opció de previsualitzar els càlculs estadístics i decidir si incorporar-los en el report o prescindir d'ells.

Per altre banda el conjunt de gràfiques que s'inclouran en l'informe, es troba definit mitjançant programació, encara que l'usuari té la possibilitat de personalitzar-ho, ja que existeix l'opció d'afegir gràfiques addicionals d'interès particular per l'usuari.

La sortida d'aquest procés, és un report en format pdf, amb la informació rellevant de la campanya, que com ja s'ha dit, conté una fitxa tècnica del projecte o campanya al que pertanyen les dades, estadístiques de la missió, gràfiques tècniques o científiques i opcionalment, les gràfiques seleccionades per l'usuari on és pot visualitzar el comportament dels diferents paràmetres.

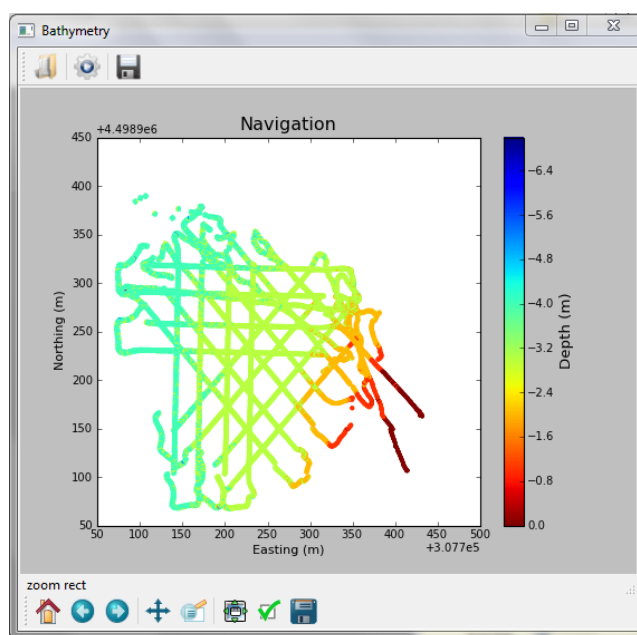
#### 4.2.4 Batimetria del terreny

Com ja s'ha mostrat anteriorment, mitjançant l'adcp incorporat a l'auv, és possible realitzar una batimetria del terreny. La finalitat d'una batimetria és l'obtenció de les coordenades XYZ dels punts submergits, i d'aquesta manera poder realitzar mapes que permeten caracteritzar i conèixer el fons marí.

L'aplicació desenvolupada implementa l'algoritme per al càlcul d'aquests punts mitjançant l'ús combinat de les dades provinents de l'adcp (arxiu.dvl) i de l'auv (arxiu.log). Planificant la trajectòria del vehicle, seria possible obtenir un conjunt

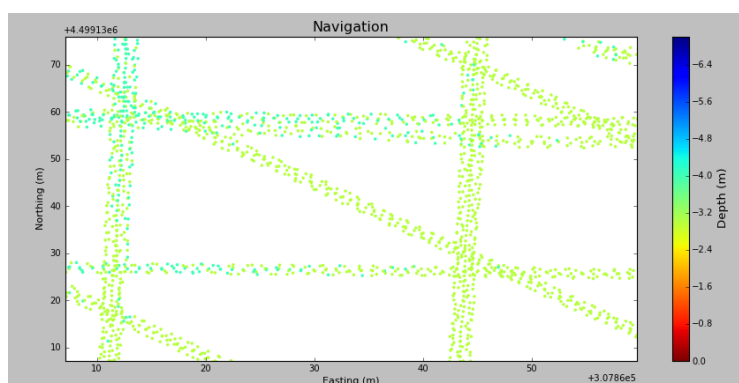
de dades batimètriques homogeni, i d'aquesta manera produir millors mapes en menys temps i menor cost utilitzant AUVs en lloc de tècniques d'estudi convencionals.

Per tant la funcionalitat de batimetria, permet processar les dades i visualitzar els resultats a través de la interfície gràfica (Fig.4.7). Un cop processades les dades, aquestes s'exporten a format ASCII amb la informació de latitud, longitud i profunditat dels punts amb les coordenades reflectides en el sistema de coordenades UTM (m) i les profunditats en metres. D'aquesta manera les dades podran ser visualitzades en altres sistemes d'informació geogràfica.



**Fig. 4.8** Interfície per generar la batimetria amb els resultats gràfics.

Fent zoom, es pot observar com la línia de la trajectòria del vehicle està formada pels diferents punts obtinguts amb els quatre feixos del ADCP.



**Fig. 4.9** Ampliació de la gràfica de resultats batimètrics.



## CAPÍTOL 5. CONCLUSIONS

L'ús de las TIC en ambients oceanogràfics involucra un guany en la certesa de la informació i en la possibilitat de la presa de decisions amb major rapidesa gràcies a la velocitat de processament que aquestes tenen.

El software desenvolupat constitueix una eina senzilla per a la gestió i control de dades, i la generació automatitzada d'informes científics i tècnics a partir de les dades obtingudes en campanyes oceanogràfiques de vehicles autònoms submarins.

El seu funcionament totalment operatiu i testejat té una aplicació directa en l'àmbit de UTM i de la seva línia d'investigació orientada al desenvolupament tecnològic i operacional dels AUVS. En particular, aquest projecte millora la gestió de la qualitat de les dades adquirides i ajuda a tenir un control sobre l'estat del vehicle, avaluant possibles disfuncions en els sensor mitjançant la visualització gràfica i numèrica de les dades. Alhora la generació d'informes de manera automatitzada permet reduir notablement el temps de realització d'aquest, i contribueix a millorar la planificació i la logística de futures missions.

Pel que fa al procés de desenvolupament, l'arquitectura del sistema ha jugat un paper fonamental dintre del desenvolupament, ja que les decisions crítiques s'han hagut de prendre des de l'inici. No crear aquest disseny, hagués pogut limitar el producte final, ficant en perill la consecució dels objectius del projecte. L'estructura de capes i la modularitat del sistema, el dota de la capacitat d'escalabilitat, deixant oberta d'aquesta manera la incorporació de nous mòduls i funcionalitats. També afegir que el haver encarat el projecte de manera modular, permetrà adaptar-lo a noves plataformes, ja que el funcionament de l'aplicació s'adequa a les característiques del model de AUV utilitzat i al tipus de fitxer que aquest genera, que depenent del fabricant pot variar bastant.

En quant al llenguatge dir que després de aprendre Python, resulta un llenguatge molt útil en l'àmbit científic, i que s'està imposant per la seva flexibilitat, potència i senzillesa. A més disposa de llibreries amb característiques similars a software propietari, com Matplotlib en el cas de Matlab, utilitzada en aquest projecte.

Un dels avantatges que es vol destacar amb la realització d'aquest projecte és l'estalvi que suposa l'ús d'eines de codi obert davant d'altres solucions. En un moment de crisis com l'actual, en què tant organismes públics com entitats privades disposen de menys recursos, l'ús de software lliure suposa un gran avantatge al no estar subjecte a llicències de pagament a diferència d'altres eines encara utilitzades per UTM com Matlab. A banda de l'estalvi econòmic, la filosofia de lliure distribució fomenta la divulgació de coneixement i permet millorar la qualitat de les aplicacions de manera col·laborativa, per tant és ideal pel món de la investigació. El software resultant és pensa distribuir sota llicència GPL (codi obert) permetent d'aquesta manera que quedi obert a futures modificacions no tan sols dintre de UTM sinó també a través de col·laboracions externes.

Tot i l'assoliment dels objectius del projecte i l'obtenció de resultats satisfactoris, la versió actual és una primera versió d'un software de gestió i control de dades oceanogràfiques realment aplicable i per tant susceptible de millores, però també un bon punt de partida per a futurs projectes i aplicacions. S'ha posat interès al respecte documentant el codi desenvolupat per facilitar-ne la comprensió i les futures modificacions.

Finalment comentar que respecte a l'impacte mediambiental de l'aplicació, remarcar que no produeix cap efecte advers en el medi ni s'ha generat cap mena de residus en el seu desenvolupament ja que tot el projecte a sigut a nivell de software. A més a més contribueix a l'ús de l'AUV com a sistema eficaç de mesura i mostreig, que complementat amb aplicacions com la desenvolupada permeten disposar d'informació d'interès sobre la salut dels oceans. D'aquesta manera és contribueix a una gestió y avaluació més adequada dels impactes que l'acció humana representa per al medi.

## 5.1 Futures millores

Com a eina desenvolupada en col·laboració amb UTM i dintre d'una de les diverses línies de treball I+D relacionades amb els propis vehicles autònoms, el desenvolupament d'aquesta eina no es dona per finalitzat i queda obert a un creixement futur i possibles millores.

A continuació s'exposen algunes de les possibles millores i futures línies a seguir:

- Suport per al format de dades NetCDF tant en lectura com en escriptura. NetCDF és un format d'arxiu estàndard autodescriptiu per al guardat de dades científiques, independent del *software* i del hardware. Actualment l'aplicació utilitza un format de dades no estandarditzat i amb l'ús de NetCDF s'aconseguiria millorar la compatibilitat i la interoperabilitat.
- Actualment no és poden guardar les configuracions de sessions anteriors, fent que cada cop que és crea un report sigui necessari configurar de nou tots els paràmetres.
- Gràfiques de les dades batimètriques obtingudes en 3D.

Desenvolupament i implementació de nous mòduls i funcionalitat com per exemple :

- **Control de qualitat de les dades:** Tenir un control de les dades i detectar punts on hi pot haver alguns problemes en l'adquisició per poder ser eliminat o corregits del fitxer de dades final. Les tècniques a utilitzar poden ser el càlcul de la variança de les dades.

- **Correcció de la navegació:** l'AUV normalment té alguns errors de posicionament quan es troba submergit ja que agafa com a referència el fons, la velocitat de navegació i un compàs magnètic intern per estimar la seva posició. L'algoritme a implementar hauria de recalculat la trajectòria tenint en compte que els punts on el vehicle es troba en superfície són correctes i obtenir els punts reals intermitjos (quan el vehicle està submergit). Com a primera aproximació suposar que l'error es lineal i després intentar tenir en compte altres factors que poden afectar en aquest error (corrents, etc.).
- **Aplicació de correcció de dades (*denoising*):** Totes les senyals obtingudes tenen soroll superposat. Programar una sèrie de filtres pel tractament de les dades i poder entregar el científic les dades "netes". Seria una possibles opcions a afegir en la interfície gràfica.

## BIBLIOGRAFIA

- [1] Gordon, L., *Acoustic Doppler current profiler - Principles of operation: A practical primer*, Second ed., RD Instruments, 1996, [PDF].
- [2] RD Instruments, *ADCP Coordinate Transformations: Formulas and Calculations*, RD Instruments, Teledyne RD Instruments, 2010, [PDF]
- [3] Dinehart, R.L. and Burau, J.R., "Repeated surveys by acoustic Doppler current profiler for flow and sediment dynamics in a tidal river", *The Journal of Hydrology*, 314(1-4), 1-21, 2005.
- [4] Kim, D., Hull, M. and Muste, M., *Fast Mapping of River and Lake Bed Shape Using Geo-referenced Bathymetry Measurements Acquired by SonTek's 9-beam ADCP*, The University of IOWA, May 2009 [PDF]. Accessible a: <http://documents.clubexpress.com/documents>
- [5] Rodríguez, P. and Piera J., "Mini AUV, a platform for future use on marine research for the Spanish Research Council", *Instrumentation ViewPoint* Issue 4, 2005. MarTech 2005 Proceedings. [PDF]
- [6] Pujol, N., Rodríguez, P. and Piera J., "New lightweight AUV at the Spanish Research Council", *Instrumentation ViewPoint*, 2011, MarTech 2011 Proceedings [PDF]
- [7] van Rossum, G., *Python Tutorial* (release 2.7.2), Python Software Foundation, 2011, [PDF].  
disponible a: <http://www.uv.es/dogarcar/man/PythonTutorial.pdf>
- [8] Summerfield, M., *Rapid GUI Programming with Python and QT: The Definitive Guide to PyQt Programming*, Prentice Hall PTR, 2007, [PDF],  
disponible a: <http://www.qtrac.eu/pyqtbook.html>
- [9] Tosi, S., *Matplotlib for Python Developers*, Packt Publishing Ltd., Birmingham, Novembre 2009, [PDF].
- [10] *ReportLab PDF Library: User Guide*, [PDF],  
disponible a: <http://www.reportlab.com/docs/reportlab-userguide.pdf>
- [11] ReportLab API Reference, [PDF],  
disponible a: <http://www.reportlab.com/docs/reportlab-reference.pdf>
- [12] Python Programming Language, Official Website [WEB]  
Accessible a l'adreça:

<http://www.python.org/>

[13] Qt Reference Documentation, Official Website [WEB]

Accessible a l'adreça:

<http://doc.qt.digia.com/qt/index.html>

[14] Matplotlib, Official Website. [WEB].

Accessible a l'adreça:

<http://matplotlib.org/>

[15] ReportLab Library, Official Website [WEB]

Accessible a l'adreça:

<http://www.reportlab.com/>

## ANNEX A: MANUAL D'USUARI

### A.1 Instal·lació

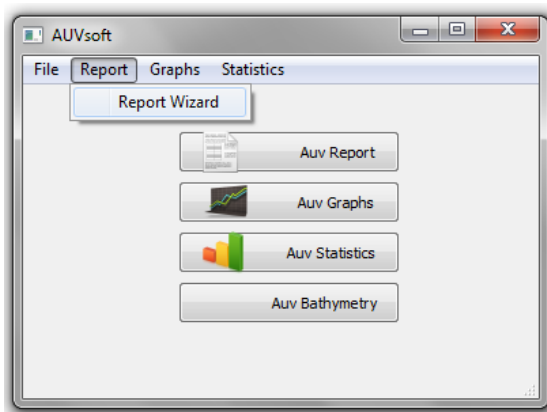
L'aplicació es distribueix mitjançant un arxiu .zip.

Aquest software, corre sota plataformes Windows 64-bit i Windows-32-bit. Per poder executar l'aplicació cal tenir instal·lades prèviament les següents llibreries:

- Python 2.7 <http://www.python.org/download/>
- PyQt4 <http://www.riverbankcomputing.co.uk/software/pyqt/download>
- Matplotlib /Numpy <https://github.com/matplotlib/matplotlib/downloads>
- Reportlab <http://www.reportlab.com/software/opensource/rl-toolkit/download/>

Per començar a utilitzar AUVsoft, descomprimir l'arxiu AUVsoft.zip i executar el fitxer mainwindow.py.

El programa s'inicialitzarà amb un menú principal, on es pot escollir entre diverses funcionalitats.



**Fig. A.1** Pantalla del menú principal de l'aplicació

- **Auv Report:** generació automatitzada d'informes científics i tècnics a partir de les dades obtingudes en campanyes oceanogràfiques de vehicles autònoms submarins (AUV).
- **Auv Graphs:** Component per a la visualització gràfica de les dades.
- **Auv Statistics:** Generació i visualització d'estadístiques bàsiques de missió, com per exemple distància recorreguda, temps de missió, màxims, mínims i mitja de diferents variables, etc.

- **Auv Bathymetry:** Generació de les coordenades x, y, z corregides dels punts d'impacte dels diferents feixos del ADCP amb el fons.

A través del menú superior, s'obre un formulari independent que permet realitzar diferents tasques de manera paral·lela.

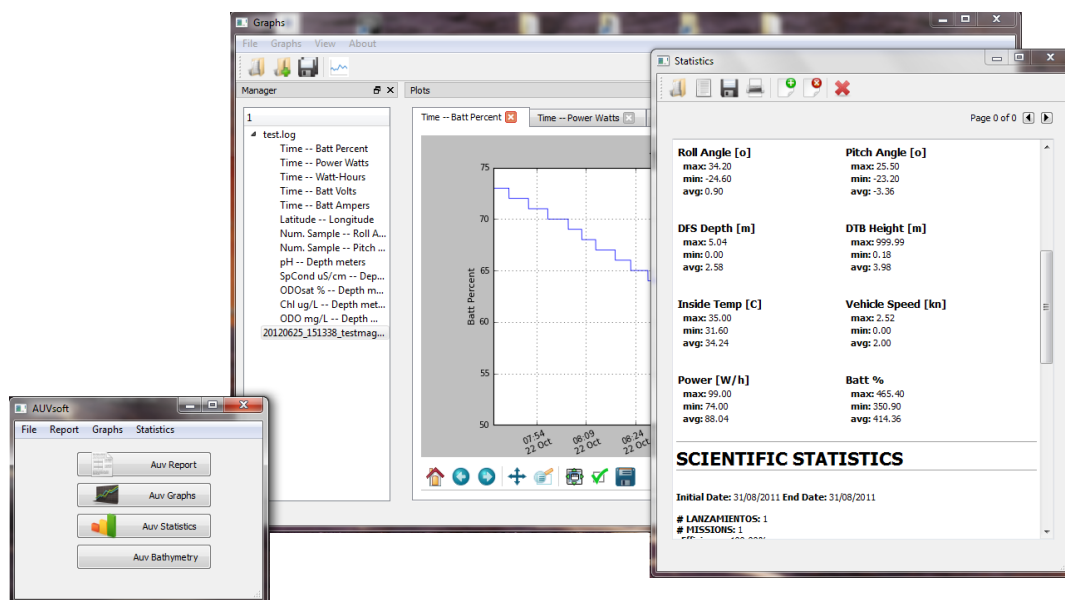


Fig. A.2 Visualització de les funcionalitats d'estadístiques i gràfiques

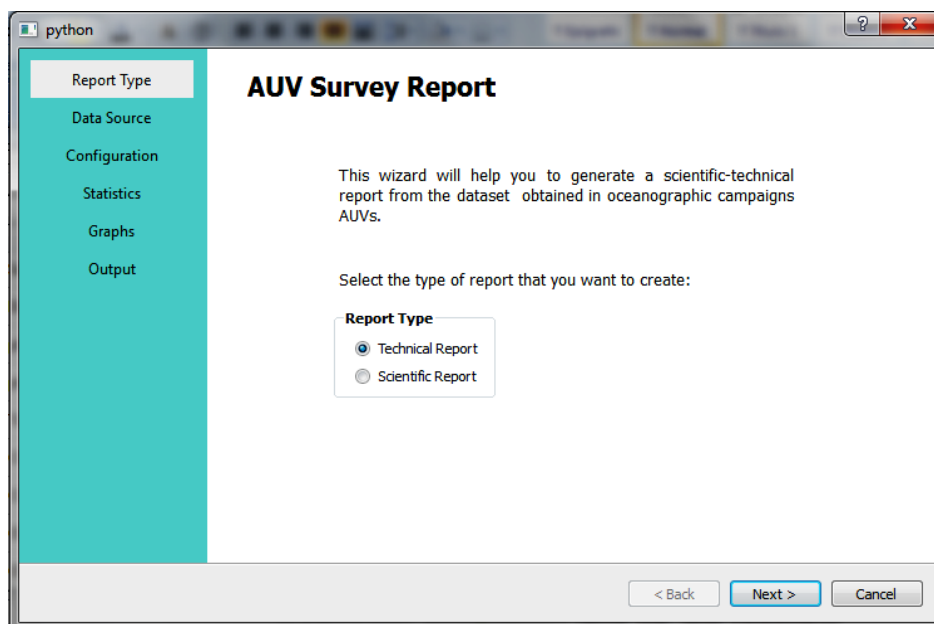
## A.II Funcionalitats

### a) Generació report en format pdf

La tasca de crear un informe de campanya es facilita gràcies a aquest assistent que simplifica el procés en cinc passes, guiant a l'usuari en tot moment.

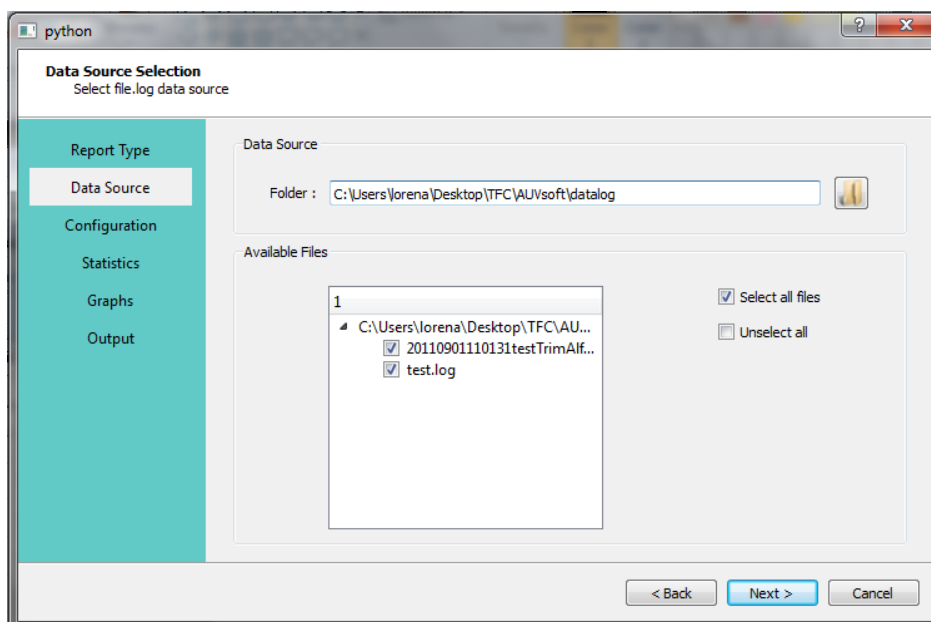
Pas 1: El primer pas és establir el tipus de report que es vol obtenir depenent de l'estudi a realitzar (Figura A.3). Aquest pot ser :

- **Tècnic:** es centra en l'anàlisi dels paràmetres d'enginyeria del vehicle, nivell i estat de les bateries, velocitat i temperatura interna del vehicle, angles roll/pitch etc.
- **Científic:** analitza les variables de tipus fisicoquímic com els nivells de clorofil·la, pH, oxigen dissolt etc.



**Fig. A.3** Pàgina 1 de l'assistent: Selecció del tipus d'informe

Pas 2 : Seleccioni el directori on es troba la font de dades. Si el directori no és correcte o no conté arxius d'extensió .log, l'aplicació llançarà un missatge d'error. Introdueixi un directori vàlid i a continuació en el quadre inferior és mostraran els fitxers .log disponibles.



**Fig. A.4** Pàgina 2 de l'assistent: Selecció dels fitxers

Pas 3: Completi la fitxa tècnica de campanya. Aquestes dades fan referència al context on s'emmarca la campanya de recollida de dades. Aquesta informació



és d'especial utilitat per referenciar la procedència de les dades analitzades. Quan tots el camps estiguin complerts, s'habilitarà el botó *Next*.

**Fig. A.5** Pàgina 3 de l'assistent: Fitxa tècnica, introducció de metadades

Pas 4: Premi el botó '*Show Statistics*' per previsualitzar els resultats estadístics dels fitxer seleccionats. Per defecte l'informe inclourà un estudi estadístic. En cas que aquest no sigui d'interès, es pot optar per no incloure'l al report desmarcant la opció ☒ **Include statistics**

Statistic	min	max	avg
DFS Depth [m]	-24.60	5.04	2.47
DTB Height [m]	-23.20	999.99	3.85
Inside Temp [C]	0.00	35.60	34.29
Vehicle Speed [kn]	0.00	4.53	1.99
Power [W/h]	74.00	99.00	87.92
Batt %	350.90	465.40	413.76

**Fig. A.6** Pàgina 4 de l'assistent: Dades estadístiques

Pas 5: Selecció de gràfiques d'usuari.

L'informe inclou un conjunt de gràfiques fixes i definides prèviament que l'usuari no pot modificar. Segons el tipus d'informe les gràfiques incloses són:

#### Gràfiques comuns

- **Gràfiques de navegació:**
  - representació de la trajectòria del vehicle en 2 i 3D.
  - gràfica de profunditats assolides vs. Nivell del fons.

#### Gràfiques específiques informe tècnic:

- **Evolució de diferents paràmetres del vehicle:**
  - Velocitat del vehicle
  - Temperatura interna
  - Magnetic compas: representació dels angles roll/pitch.
- **Lectura dels nivells de bateria:**
  - Batt (V) i Batt(A)
  - Potencia (W)
  - Consum energia (kWh) i Nivell de Bateria %

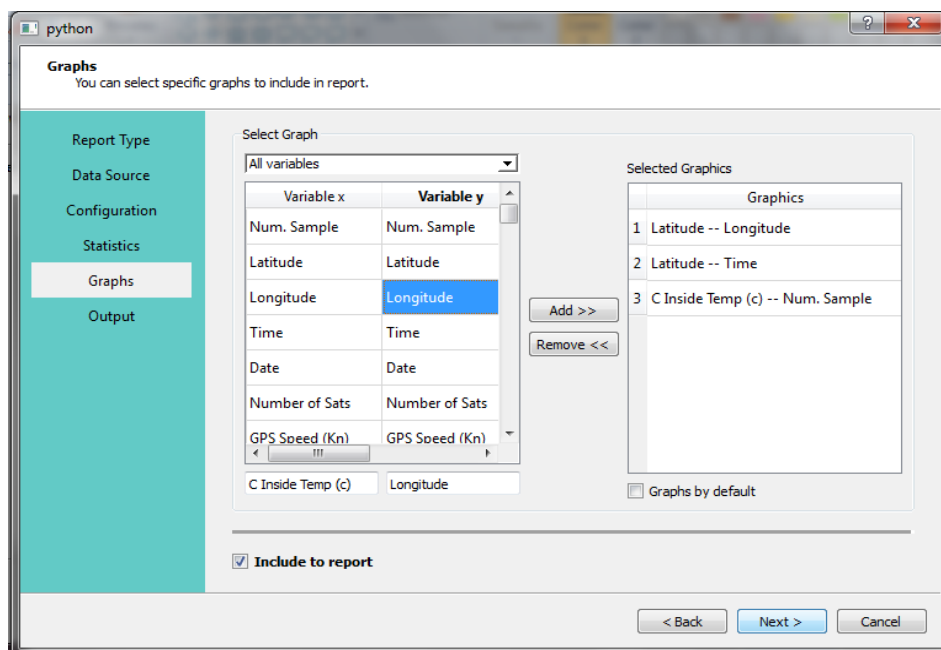
#### Gràfiques específiques informe científic:

- **Perfil vertical:** representació de diferents paràmetres fisicoquímics en funció de la profunditat.
  - Temperatura, pH, Conductivitat, Salinitat, Nivell de clorofil·la, Oxigen dissolt i terbolesa.

No obstant això, si necessita incloure altres gràfiques a l'informe, l'assistent ofereix la possibilitat de crear la seva llista per afegir-la al conjunt de gràfiques ja esmentat. (Fig. A.7 ). Per crear la llista de gràfiques:

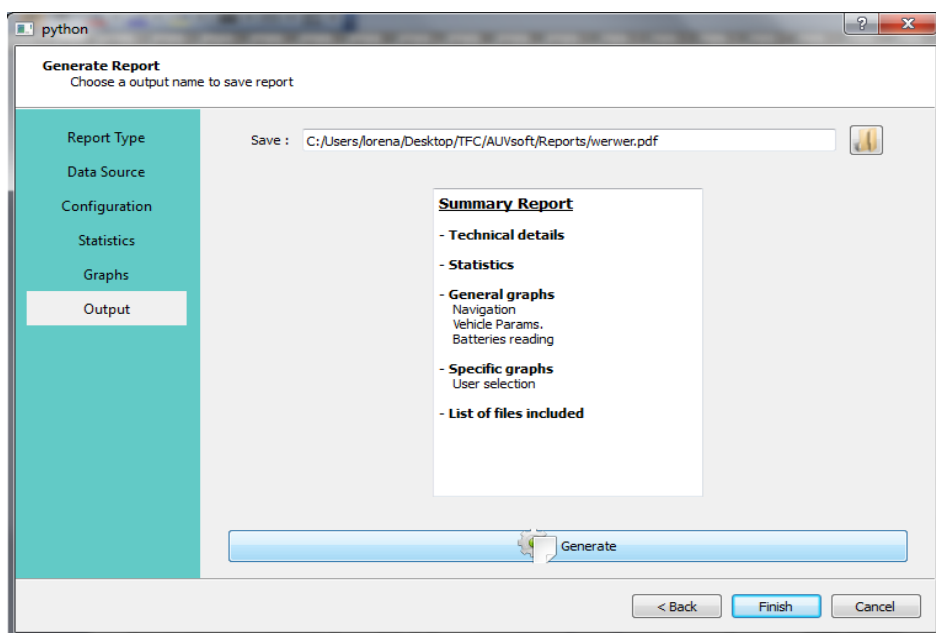
- En la part superior de la taula amb les variables (taula esquerra), pot seleccionar *All variables*, per observar tots els paràmetres disponibles o *Scientific variables*, que mostrarà una taula reduïda, només amb els paràmetres de tipus fisicoquímic.
- Seleccioni el conjunt x,y en la taula clicant a sobre de la variable desitjada. En la part inferior de la taula, pot observar les variables que selecciona per a x i y. A continuació polsi el botó Add.
- La gràfica seleccionada apareixerà en la taula de la dreta.
- L'aplicació llançarà diferents missatges si realitza una operació incorrecta com per exemple afegir una gràfica existent, escollir una parella x-y no possible (per exemple, data en funció del temps) o si x e y és la mateixa variable.

- Un cop creada la taula de gràfiques activi ☒ Include to report



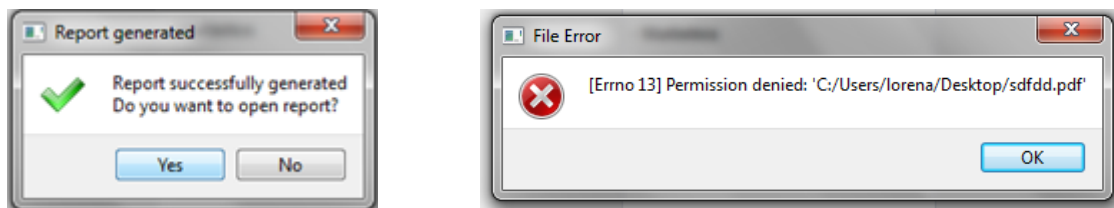
**Fig. A.7** Pàgina 5 de l'assistent: Selecció de les gràfiques d'usuari.

Pas 6: Per finalitzar amb la configuració del report, seleccioni el directori i un nom per l'arxiu de sortida. La finestra de l'assistent mostrarà un índex amb el contingut l'informe (Fig A.8). Premi el botó *Generate* si està d'acord o torni enrere per modificar els paràmetres de configuració.



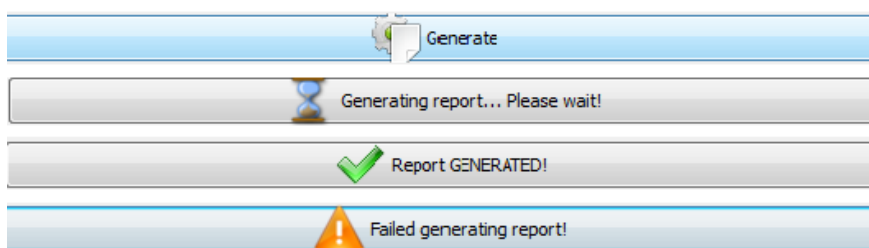
**Fig. A.8** Pàgina 6 de l'assistent: Generació de l'informe.

Esperi mentre és realitza el procés. Al finalitzar la creació de l'informe amb èxit, apareix una finestra que li dona l'opció d'obrir directament l'informe creat. Si succeeix algun error creant el pdf o obrint el fitxer pdf, l'aplicació llançarà una excepció i informará de l'error ocorregut mitjançant una finestra.



**Fig. A.9** Missatges finals del procés de report.

Per altre banda, el botó *Generate* (Fig. A.8) anirà canviant per informar-li en quin estat del procés es troba.

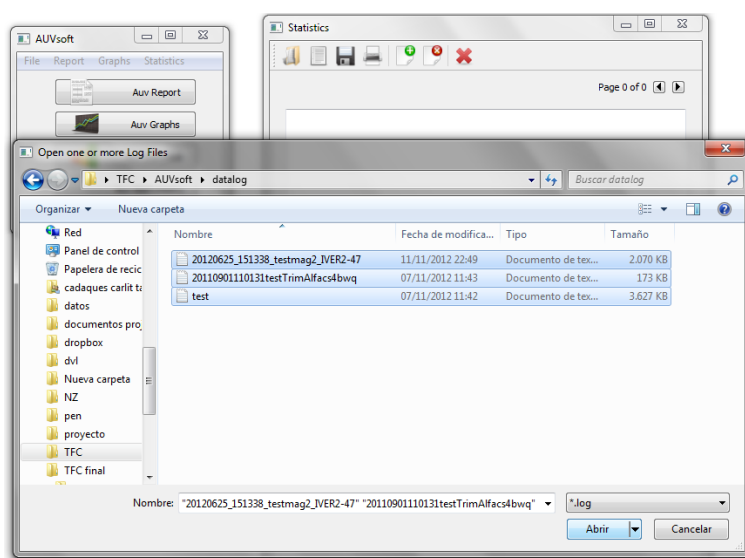


**Fig. A.10** Exemples d'informació del botó *Generate* (pas 6 de l'assistent)

## b) Generació d'estadístiques

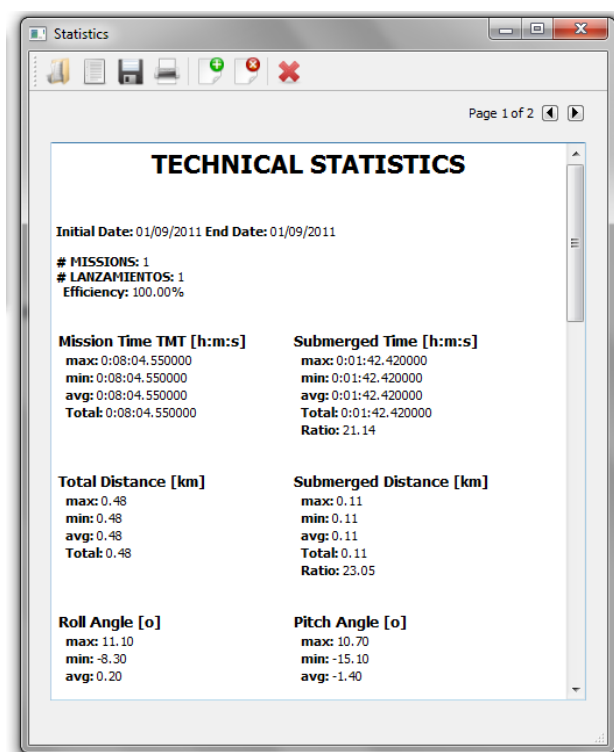
La funcionalitat d'estadístiques li permet generar un conjunt d'estadístiques bàsiques de missió a partir d'un fitxer de dades .log. Podrà visualitzar-les a la pantalla principal, exportar-les a un fitxer .txt o pdf o imprimir-les.

- Per generar una nova estadística, seleccioni la icona carpeta en la barra d'eines superior. S'obrirà la finestra de diàleg '*Open one or more Log Files*' (Fig. A.11) on podrà accedir a les unitats de xarxa o disc disponibles en el seu sistema.
- Des de la llista de directoris, seleccioni la carpeta que conté els arxius de dades .log. A continuació seleccioni un o diversos arxius. La selecció de més d'un arxiu, genera estadístiques de tot el conjunt de dades inclosos als diferents fitxers.



**Fig. A.11** Diàleg de selecció dels fitxers

Un cop carregats els fitxers seleccionats, apareixeran els resultats en la pantalla principal, tants les estadístiques tècniques com les científiques.



**Fig. A.12** Pantalla principal del bloc d'estadístiques amb els resultats

El menú superior permet realitzar les següents operacions:



- Generar una nova estadística a partir d'un o més fitxers d'estat del vehicle d'extensió .log.



- Carregar una estadística existent a partir d'un fitxer .txt



- Guardar els resultats en un arxiu d'extensió .txt



- Imprimir els resultats. Quan es realitza l'operació d'imprimir, alhora permet guardar els resultats en un arxiu .pdf.



- Afegir una nova pàgina per generar noves estadístiques . Aquesta opció permet visualitzar estadístiques de diferents fitxers simultàniament mitjançant l'ús de diverses pàgines.





- Eliminar una pàgina creada.

A banda d'això, també es poden realitzar les tasques bàsiques d'un editor de text, com copiar, pegar etc. amb un clic dret del ratolí.

### c) Batimetria

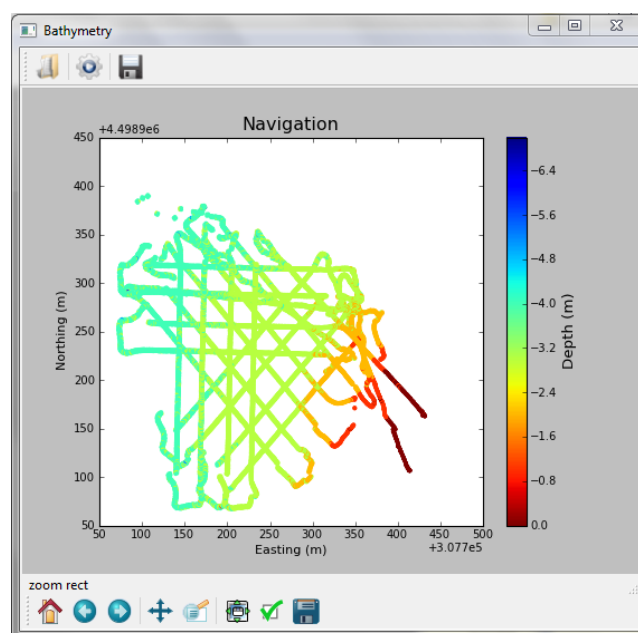
La funcionalitat de batimetria permet generar les coordenades dels punts d'intersecció dels feixos del dvl/adcp amb el fons, i d'aquesta manera obtenir un conjunt de punts per representar el fons marí. Els resultats s'exporten a un fitxer .txt per a la posterior visualització.

Per processar les dades emmagatzemades en els fitxers .log i .dvl :

- Seleccioni la icona  en la barra d'eines (Fig. A.13). A continuació seleccioni un fitxer d'extensió .dvl en el quadre de diàleg. Per generar els resultats, el directori on es troba aquest fitxer també ha de contenir un altre fitxer amb el mateix nom, però d'extensió .log. Premi acceptar
- Si els fitxers es carreguen correctament, seleccioni la icona  per generar els resultats.

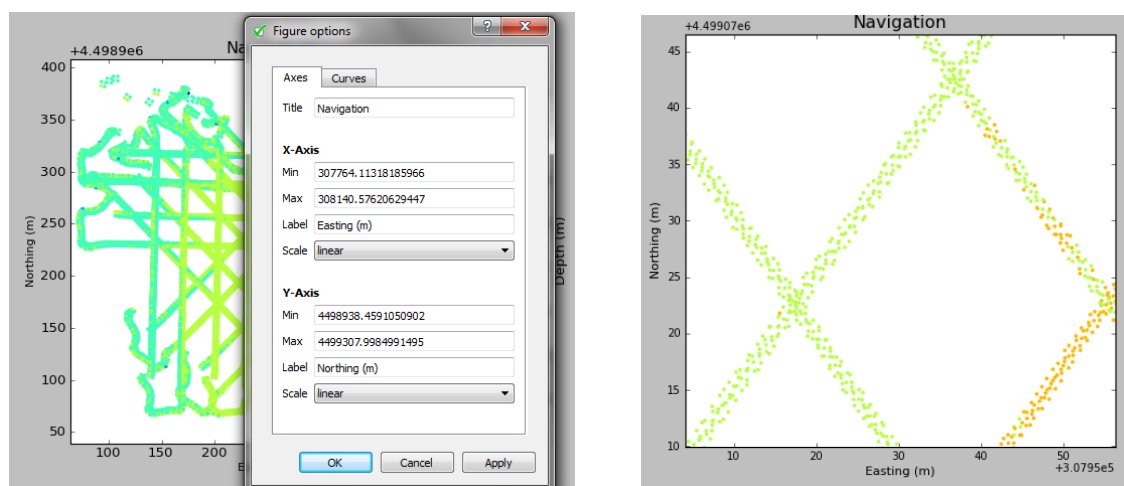
Un cop processades les dades satisfactòriament, s'obrirà un quadre de diàleg on ha de seleccionar el nom i el directori de l'arxiu de sortida. En aquest fitxer s'emmagatzemaran els resultats amb la informació de latitud, longitud i profunditat dels punts (Coordenades en UTM i profunditats en metres).

Després de crear el fitxer de resultats, a la pantalla principal es mostrarà la representació de les coordenades i profunditat per a cada punt. (Fig. A.13)



**Fig. A.13** Finestra per a la generació de la batimetria.

La gràfica generada conté una barra de navegació per realitzar diverses operacions sobre la gràfica. És pot guardar, ampliar, veure les coordenades d'un punt i canviar les propietats de la figura i dels eixos (Fig. A.14)





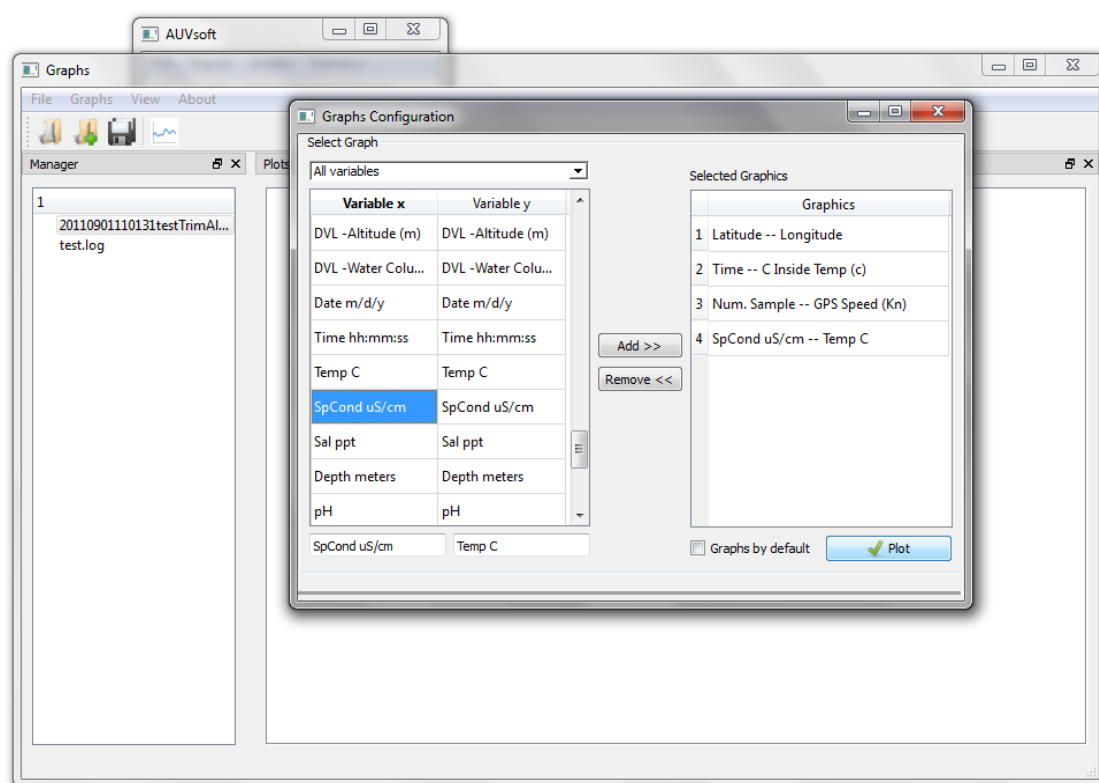
**Fig. A.14** Propietats de la figura i zoom de la imatge

### d) Generació de gràfiques

El component gràfic de l'aplicació permet visualitzar l'evolució dels diferents paràmetres continguts al fitxer d'estat del vehicle .log.

Per generar diferents gràfiques d'un fitxer:

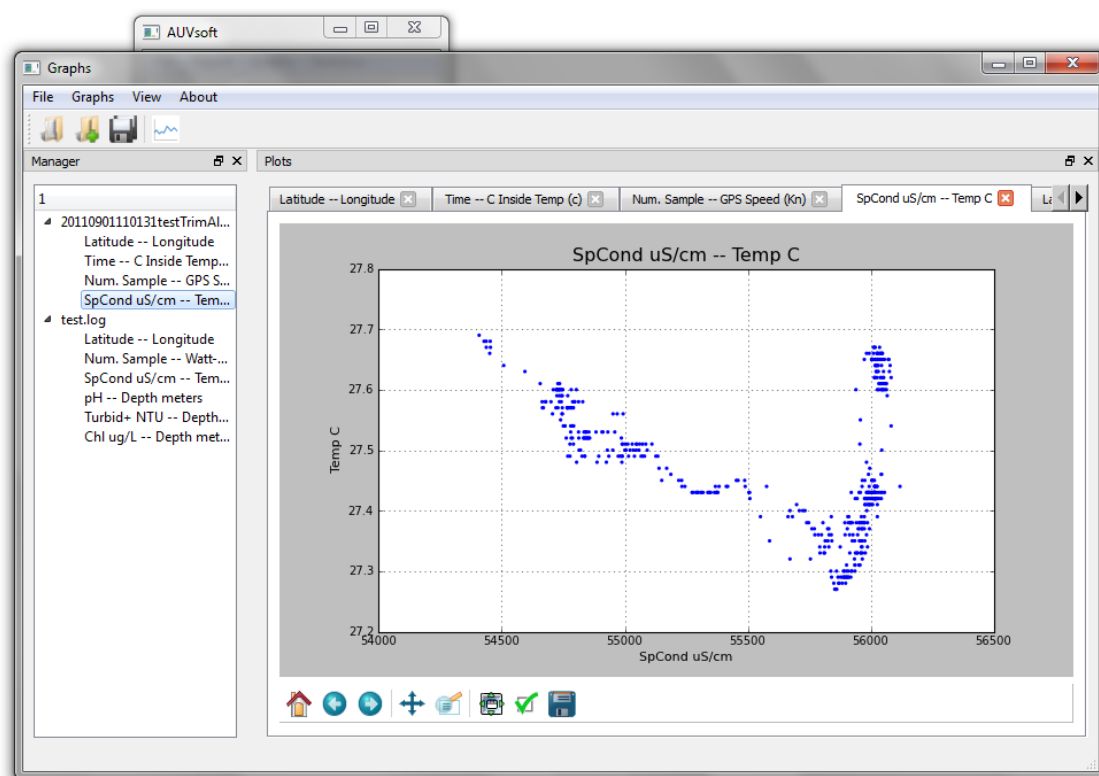
- Seleccioni la font de dades a través de la icona  . Seleccioni un o diversos fitxers dels que s'extrauran les diferents variables a representar.
- Un cop carregats els fitxers, ha de triar les variables a representar, creant una llista de gràfiques. Per escollir les gràfiques d'interès, primer marqui el fitxer en el gestor d'arxius o Manager (a l'esquerra de la pantalla) del qual vol fer les representacions. Seguidament accedeixi a la llista de variables disponibles mitjançant la icona  . Figura A.15



**Fig. A.15** Formulari de selecció de variables

Després de crear la llista de gràfiques, premi el botó *plot* per representar-les. En la finestra principal de l'aplicació, es visualitzarà el conjunt de gràfiques del fitxer seleccionat . Cada gràfica és presenta de manera individual en les diverses pestanyes i compta amb una barra de navegació mitjançant la qual pot realitzar diverses operacions sobre la imatge, com guardar, ampliar, mostrar les coordenades d'un punt, canviar el tipus de representació etc. Figura A.16





**Fig. A.16** Pantalla principal per a la representació gràfica

Mitjançant una estructura en arbre permet visualitzar el nombre de gràfiques creades i l'arxiu al qual pertany, d'aquesta manera pot tenir un control sobre el que està visualitzant. Al clicar sobre una de les gràfiques en el gestor d'arxius, s'actualitzarà la selecció a les pestanyes per mostrar la gràfica seleccionada.

Altres opcions que pot realitzar són:

- Eliminar una gràfica
- Guardar totes les gràfiques de cop: Ha de seleccionar el nom d'un directori on s'emmagatzemaran totes les imatges que hagi creat.
- Guardar les gràfiques de manera individual.
- Ocultar el gestor d'arxius o expandir la zona de les gràfiques. Inclús pot arrossegar la part que conté les gràfiques fora de la finestra principal, i d'aquesta manera veure-les a pantalla completa.

### A.III Tipus de fitxers

A continuació és mostra un exemple de les tres primeres línies del fitxer d'estat del vehicle i del dvl/adcp.

#### Arxiu dvl

```
Latitude;Longitude;Time;Date;C True Heading;Sample Number;Fix Type;
Fix Quality;X Speed(m/s);Y Speed(m/s);Z Speed(m/s);Track Time;
XDistance(m);YDistance(m);Altitude(m);Depth(m);Temperature(c);
SoundSpeed(m/s);Salinity(ppt);BtXspd(m/s);BtYspd(m/s);BtZspd(m/s);B
tQuality;BtRnge1(m);BtRnge2(m);BtRnge3(m);BtRnge4(m);BtXWaterSpd(m/
s);BtYWaterSpd(m/s);BtZWaterSpd(m/s);BtWaterQuality;Xvelocity(m/s);
Yvelocity(m/s)
```

```
40.61962;0.731485;15:13:44.77;6/25/2012;306.499999952316;41;0;0;0.0
56;0.017;0.045;40;-21.2;0.95;0.52;9999.99;30.5;1547.6;36;-
0.056;0.017;0.045;9;
0.53;0.52;0.54;0.54;999.999;999.999;999.999;0;0.019;-0.095
```

```
40.61962;0.731485;15:13:45.79;6/25/2012;312.699999952316;42;2;9;0.0
37;-0.021;-0.039;41;-21.64;1.19;0.49;9999.99;30.5;1547.6;36;-
0.037;0.021;-
0.039;9;0.49;0.49;0.5;0.54;999.999;999.999;999.999;0;0.312;-0.359
```

#### Arxiu .log





```
Latitude;Longitude;Time;Date;Number of Sats;GPS Speed (Kn);GPS True
Heading;GPS Magnetic Variation;HDOP;C Magnetic Heading;C True
Heading;Pitch Angle;Roll Angle;C Inside Temp (c);DFS Depth (m);DTB
Height (m);Total Water Column (m);Batt Percent;Power Watts;Watt-
Hours;Batt Volts;Batt Ampers;Batt State;Time to Empty;Current
Step;Dist To Next (m);Next Speed (kn);Vehicle Speed (kn);Motor
Speed CMD;Next Heading;Next Long;Next Lat;Next Depth (m);Depth Goal
(m);Vehicle State;Error State;Fin Pitch R;Fin Pitch L;Pitch
Goal;Fin Yaw T;Fin Yaw B;Yaw Goal;Fin Roll;DVL-Depth (m);DVL -
Altitude (m);DVL -Water Column (m);DVL-FixType;DVL-FixQuality;Temp
C;SpCond uS/cm;Sal ppt;Depth meters;pH;pH mV;Turbid+ NTU;Chl
ug/L;Chl RFU;BGA-PE cells/mL;ODOsat %;ODO mg/L;Battery volts;
```





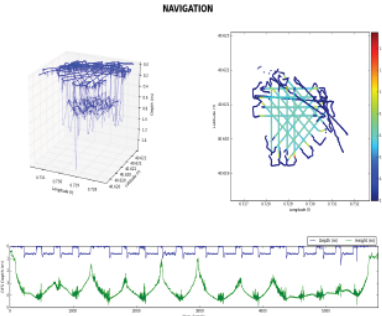
```
40.61962;0.731485;15:13:44.77;6/25/2012;7;0.18;244.54;-
3.20000004768372;1.1;309.7;306.499999952316;0.6;-
1;35.4;.00;.52;.52;87;30;403.0;15.9;-
1.9;D;879.1;0;243.730;2.5;0;128;334.4169;0.730238;40.621597;0;0;1;N
;71;71;15;64;64;275.0312;0;9999.99;.52;0.647;0;0;30.18;63905;43.02;
0.100;8.20;-81.1;1.4;7.3;1.7;11663;106.3;6.33;11.8;
```

```
40.61962;0.731485;15:13:45.79;6/25/2012;7;0.18;244.54;-
3.20000004768372;1.1;315.9;312.699999952316;1.8;-
2;35.4;.01;.49;.50;87;30;403.0;15.9;-
1.9;D;876.8;0;243.730;2.5;0;128;334.4169;0.730238;40.621597;0;0;1;N
;67;67;15;49;49;275.0312;0;9999.99;.49;0.617;2;9;30.17;63902;43.02;
0.113;8.20;-81.1;0.6;0.6;0.1;13738;106.3;6.32;11.8;
```

## ANNEX B: REPORT TÈCNIC

Exemple d'un informe generat amb el software.

 		 	
<h3>TECHNICAL REPORT</h3> <p>Laboratorio de Plataformas Autónomas UTM-CSIC</p>		Project technical details	
		Project Name: This is a Test Acronym: This is a Test Project Ref.: This is a Test Organization: This is a Test Initial Date: 01/01/2012 End Date: 01/01/2012	
		Abstract	
		This is a Test	
		Key Words: This is a Test	
Pag. 2			

 		 													
Survey Statistics		General graphs													
INITIAL DATE: 21/10/2011 END DATE: 25/06/2012 # MISSIONS: 2 # LANZAMIENTOS: 2 EFFICIENCY: 100.00%		General graphs - File 0: /data/log/20120625_151338_testmag2_VER2-47.log													
<table border="0"> <tr> <td> <b>MISSION TIME TMT [h:m:s]</b>            Max: 1:37:31.040000            Min: 0:58:54.310000            Avg: 1:18:12.675000            Total: 2:36:25.350000         </td> <td> <b>SUBMERGED TIME [h:m:s]</b>            Max: 0:55:38.460000            Min: 0:36:46.600000            Avg: 0:46:12.530000            Total: 1:32:25.060000            Ratio: 59.08         </td> </tr> <tr> <td> <b>TOTAL DISTANCE [km]</b>            Max: 7.21            Min: 3.80            Avg: 5.50            Total: 11.01         </td> <td> <b>SUBMERGED DISTANCE [km]</b>            Max: 3.58            Min: 2.33            Avg: 2.96            Total: 5.91            Ratio: 53.71%         </td> </tr> <tr> <td> <b>ROLL ANGLE [°]</b>            Max: 54.20            Min: -32.10            Avg: 4.91         </td> <td> <b>PITCH ANGLE [°]</b>            Max: 26.70            Min: -31.20            Avg: -1.18         </td> </tr> <tr> <td> <b>DFS DEPTH [m]</b>            Max: 15.43            Min: 0.00            Avg: 2.97         </td> <td> <b>DTB HEIGHT [m]</b>            Max: 19.57            Min: 0.32            Avg: 5.51         </td> </tr> <tr> <td> <b>INSIDE TEMP [°C]</b>            Max: 35.60            Min: 22.10            Avg: 29.71         </td> <td> <b>VEHICLE SPEED [kn]</b>            Max: 3.28            Min: 0.00            Avg: 5.51         </td> </tr> <tr> <td> <b>POWER [W/h]</b>            Max: 403.00            Min: 271.70            Avg: 333.73         </td> <td> <b>BATT [%]</b>            Max: 87.00            Min: 57.00            Avg: 71.57         </td> </tr> </table>		<b>MISSION TIME TMT [h:m:s]</b> Max: 1:37:31.040000 Min: 0:58:54.310000 Avg: 1:18:12.675000 Total: 2:36:25.350000	<b>SUBMERGED TIME [h:m:s]</b> Max: 0:55:38.460000 Min: 0:36:46.600000 Avg: 0:46:12.530000 Total: 1:32:25.060000 Ratio: 59.08	<b>TOTAL DISTANCE [km]</b> Max: 7.21 Min: 3.80 Avg: 5.50 Total: 11.01	<b>SUBMERGED DISTANCE [km]</b> Max: 3.58 Min: 2.33 Avg: 2.96 Total: 5.91 Ratio: 53.71%	<b>ROLL ANGLE [°]</b> Max: 54.20 Min: -32.10 Avg: 4.91	<b>PITCH ANGLE [°]</b> Max: 26.70 Min: -31.20 Avg: -1.18	<b>DFS DEPTH [m]</b> Max: 15.43 Min: 0.00 Avg: 2.97	<b>DTB HEIGHT [m]</b> Max: 19.57 Min: 0.32 Avg: 5.51	<b>INSIDE TEMP [°C]</b> Max: 35.60 Min: 22.10 Avg: 29.71	<b>VEHICLE SPEED [kn]</b> Max: 3.28 Min: 0.00 Avg: 5.51	<b>POWER [W/h]</b> Max: 403.00 Min: 271.70 Avg: 333.73	<b>BATT [%]</b> Max: 87.00 Min: 57.00 Avg: 71.57	 <p>Figure 1: Navigation</p>	
<b>MISSION TIME TMT [h:m:s]</b> Max: 1:37:31.040000 Min: 0:58:54.310000 Avg: 1:18:12.675000 Total: 2:36:25.350000	<b>SUBMERGED TIME [h:m:s]</b> Max: 0:55:38.460000 Min: 0:36:46.600000 Avg: 0:46:12.530000 Total: 1:32:25.060000 Ratio: 59.08														
<b>TOTAL DISTANCE [km]</b> Max: 7.21 Min: 3.80 Avg: 5.50 Total: 11.01	<b>SUBMERGED DISTANCE [km]</b> Max: 3.58 Min: 2.33 Avg: 2.96 Total: 5.91 Ratio: 53.71%														
<b>ROLL ANGLE [°]</b> Max: 54.20 Min: -32.10 Avg: 4.91	<b>PITCH ANGLE [°]</b> Max: 26.70 Min: -31.20 Avg: -1.18														
<b>DFS DEPTH [m]</b> Max: 15.43 Min: 0.00 Avg: 2.97	<b>DTB HEIGHT [m]</b> Max: 19.57 Min: 0.32 Avg: 5.51														
<b>INSIDE TEMP [°C]</b> Max: 35.60 Min: 22.10 Avg: 29.71	<b>VEHICLE SPEED [kn]</b> Max: 3.28 Min: 0.00 Avg: 5.51														
<b>POWER [W/h]</b> Max: 403.00 Min: 271.70 Avg: 333.73	<b>BATT [%]</b> Max: 87.00 Min: 57.00 Avg: 71.57														
Pag. 3		Pag. 4													

